



ESCUELA TÉCNICA SUPERIOR DE INGENIEROS INDUSTRIALES Y DE TELECOMUNICACIÓN

Titulación :

INGENIERO TÉCNICO DE TELECOMUNICACIÓN,
ESPECIALIDAD EN SONIDO E IMAGEN

Título del proyecto:

“DETERMINACIÓN DE LAS PREFERENCIAS EN EL
DISEÑO DE COLOR DE PÁGINAS WEB”

Gonzalo Rocamador Murillo

Carlos Sáenz Gamasa

Pamplona, 1 de Julio del 2010

ÍNDICE

1	INTRODUCCIÓN	1
2	OBJETIVO	1
3	ANTECEDENTES	2
3.1	Color.	2
3.1.1	Luz	2
3.1.2	Visión del color	3
3.1.3	Atributos del color	4
3.1.4	Color en monitores	5
3.1.5	El color en la Web	5
3.1.6	Psicología del color.	5
3.2	Experimentos en páginas Web	7
3.3	Ordenación	8
3.3.1	Algoritmo mergesort	9
4	MATERIALES Y MÉTODO	15
4.1	Herramientas	15
4.1.1	HTML	15
4.1.2	PHP	15
4.1.3	SQL	16
4.1.4	Servidor y Base de datos	16
4.1.5	phpMyAdmin	17
4.1.6	Software	17
4.2	Conjunto de imágenes	18
4.2.1	Clasificación según criterios	20
4.3	Métodos estadísticos de análisis	24
4.3.1	Introducción	24
4.3.2	Prueba de Friedman	25
4.3.3	Prueba de Wilcoxon	25
4.3.4	Prueba de Mann-Whitney	27

5	APLICACIÓN	28
5.1	Descripción General	28
5.2	Esquema aplicación Web	29
5.3	Aspecto visual de la aplicación	30
5.3.1	Índice	30
5.3.2	Portada	31
5.3.3	Instrucciones	32
5.3.4	Formulario	33
5.3.5	Experimento	34
5.3.6	Feedback	35
5.4	Almacenamiento	36
5.4.1	Lista de fotos	37
5.4.2	Base de datos	38
5.5	Funcionamiento de la aplicación	42
5.5.1	Idiomas	42
5.5.2	Portada e Instrucciones	45
5.5.3	Formulario	46
5.5.4	Experimento	52
5.5.5	Feedback	59
6	RESULTADOS	63
6.1	Datos sin procesar	63
6.2	Filtrado	64
6.2.1	Tiempo de respuesta	65
6.2.2	Sin datos estadísticos	68
6.2.3	Deficiencias en la visión del color.	68
6.2.4	Estadísticos descriptivos después del filtrado	69
6.3	Resultados de la ordenación de las páginas Web.	69
6.4	Análisis según criterios	75
6.4.1	Según claro-oscuro	75
6.4.2	Según el fondo	79
6.4.3	Según el tono	83
6.4.4	Conclusiones generales de los resultados	86
7	CONCLUSIONES	87

Anexos

1.Fuentes	91
1.1 Bibliografía	91
1.2 Webgrafía	91
2.Código	92
2.1 Index.php	92
2.2 Portada.php	94
2.3 Instrucciones	95
2.3.1 Instrucciones.php	95
2.3.2 texto.php	97
2.3.3 idiomas.php	102
2.4 Formulario	102
2.4.1 Formulario.php	102
2.4.2 Conexion.php	106
2.4.3 Cogelista.php	106
2.4.4 Media.php	107
2.5 Iterativo	108
2.5.1 Iterativo.php	108
2.5.2 Recogedatos.php	108
2.5.3 Mandados.php	110
2.5.4 Biblioteca.php	112
2.5.5 Tiempo.php	115
2.5.6 Imagenes.php	116
2.6 Feedback	116
2.6.1 Feedback.php	116
2.6.2 Vistasypuntos.php	117
2.6.3 Meter.php	118
2.6.4 Infousuario.php	118
2.6.5 Seleccionusuario.php	119
2.6.6 Nuevofeed.php	119
3.Otros criterios de selección de Imágenes	122
3.1 Monocromáticas	123
3.2 Grises con diferentes tonos	124
3.3 Negras y tonos	124
3.4 Blancas y tonos	125
3.5 Tonos opuestos	125
3.6 Tonos Cálidos	126
3.7 Tonos Fríos	126

1 INTRODUCCIÓN

El color es un componente de diseño importante en gran número de aplicaciones. Las connotaciones subjetivas del color y de las combinaciones de colores se traducen en patrones de preferencia concretos, que dependen de cada sujeto. Este hecho es aprovechado por artistas y diseñadores gráficos entre otros pero esta presente en otras muchas aplicaciones y actividades.

En el gran mercado que supone Internet es muy importante realizar un sitio con diseño atractivo. Las grandes empresas lo saben e invierten importante capital en diseñar un sitio a medida de sus potenciales usuarios. La combinación cromática que se elija es entonces un factor clave para comenzar el diseño. Las preguntas que surgen son numerosas, ¿Qué combinaciones cromáticas gustan mas a los usuarios? ¿Que color gusta mas?

Las respuestas a estas preguntas son complicadas, puesto que la sensación a cada color es muy subjetiva y depende de numerosos factores físicos y culturales; la mejor manera de contestarlas es llevar a cabo un estudio con una muestra representativa y tratar de buscar patrones en las preferencias de los usuarios.

Este proyecto pretende analizar las preferencias cromáticas de los usuarios en las páginas Web. Para ello se utiliza como medio el propio Internet diseñando una aplicación en la que los usuarios establezcan su preferencia entre diferentes imágenes correspondientes a sitios Web de diferente diseño cromático.

Las conclusiones que se puedan extraer de este estudio supondrán una gran ayuda a la hora de elegir una combinación cromática para el diseño de una Web, eligiendo esta combinación en función del tipo de público que se quiera dirigir o la sensación que quiera producir en dicho público.

2 OBJETIVO

El objetivo del presente trabajo de fin de carrera es la determinación de las preferencias del usuario respecto del diseño de color de las páginas y sitios Web. Con objeto de recopilar datos de una manera eficiente se diseñará y pondrá a punto una aplicación informática que permita acceder al experimento psicofísico en entorno Web.

El experimento se realiza a través de una aplicación Web, en esta se presenta al usuario una serie de parejas de imágenes asociadas a páginas Web reales con diferentes diseños cromáticos, el usuario elige de cada pareja la que mas le gusta basándose únicamente en el diseño cromático.

De cada usuario se obtendrán una serie de datos estadísticos, tras conseguir una muestra considerable de usuarios se analizarán los resultados y se sacarán conclusiones.

3 ANTECEDENTES

3.1 Color.

El concepto de color va asociado intrínsecamente a unas características tanto físicas como psicológicas. Las diferentes longitudes de ondas emitidas por una luz generan un estímulo en el observador, este estímulo es procesado y convertido en una sensación en el cerebro, esta sensación es la que denominamos como color.

La reflexión de la luz en los objetos o una fuente de luz generan en el ojo del observador un estímulo, las diferentes longitudes de onda de dicha luz son sintetizadas en el cerebro asociándolas a un color concreto.

En un primer paso para comprender el concepto de color es necesario conocer la fuente que lo produce, esta fuente es la luz.

3.1.1 Luz

El espectro electromagnético es la distribución de energía del conjunto de todas las ondas electromagnéticas. La luz visible es una pequeña parte del espectro electromagnético, es la parte visible al ser humano de dicho espectro.

El espectro está clasificado en función de las diferentes longitudes de ondas que puedan tomar las ondas electromagnéticas; La luz se encuentra en el intervalo de longitudes de onda de 380 a 780nm.

En la Fig.1 se muestra el espectro electromagnético y dentro de él marcado las longitudes de ondas correspondientes a la luz visible y a sus colores asociados.

La fuente más común emisora de luz es el sol. La distribución espectral de la luz diurna también puede variar dependiendo de la altitud del sol o de factores climáticos.

El espectro de la luz emisora afecta directamente al color de los objetos, con una luz verde, todos los objetos tendrán un tono verdoso; tampoco es el mismo color el rojo de una manzana a plena luz del sol que si se observará al anochecer.

Existen numerosos tipos de fuentes de luz, todas ellas tienen una composición cromática diferente que otorgarán un color diferente al objeto iluminado. Por ejemplo la luz de una bombilla de tungsteno enfatiza los colores cálidos de los objetos a los que ilumina.

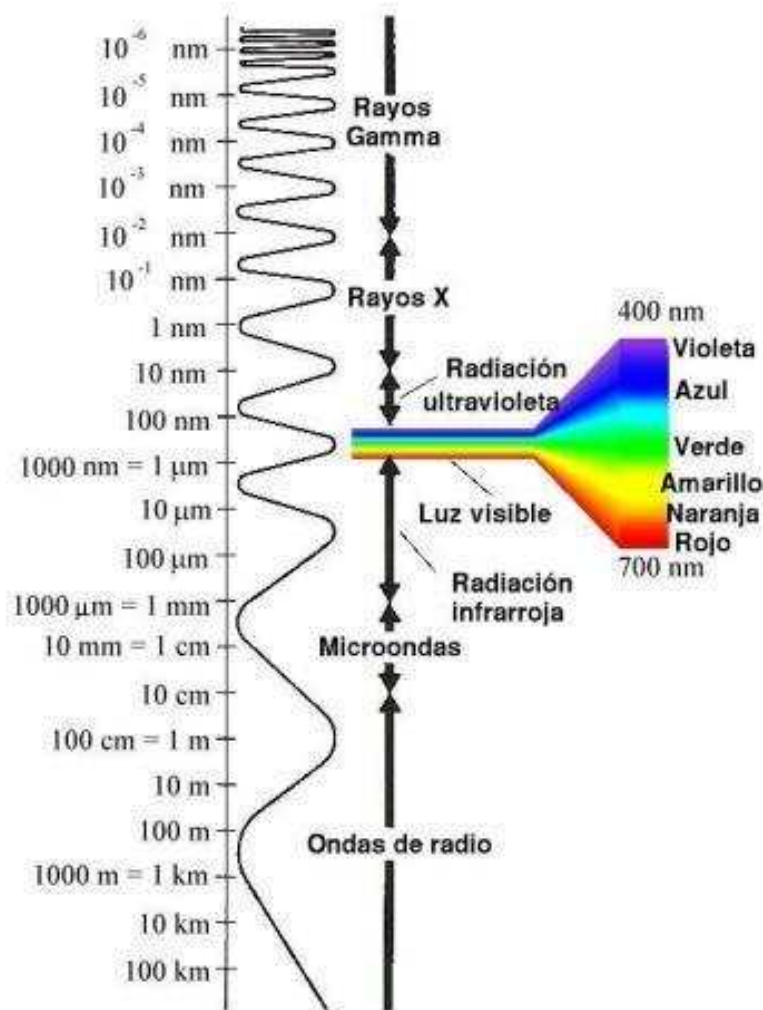


Fig.1

3.1.2 Visión del color

La visión es el sentido de captar luz e interpretarla. Este sentido se realiza mediante el sistema visual. El sistema visual está compuesto por los ojos y el cerebro, los primeros son los receptores de la luz y el segundo el encargado de interpretar la señal.

El sistema visual humano comienza en la retina. La retina es un tejido sensible a la luz situado en la superficie interior del ojo. Es similar a una tela donde se proyectan las imágenes. La luz que incide en la retina desencadena una serie de fenómenos químicos y eléctricos que finalmente se traducen en impulsos nerviosos que son enviadas hacia el cerebro por el nervio óptico.

Dentro de la retina se encuentran dos tipos de células fotorreceptoras, los conos y los bastones.

Los bastones responden a niveles de baja luminosidad, se encargan por tanto de la visión nocturna. Se encuentran en la zona alejada de la fovea, se saturan en condiciones de alta luminosidad.

Los conos se concentran en la fovea, son también sensibles a la luz y responsables de los colores, existen tres tipos de conos en el ser humano cada uno es sensible a una determinada longitud de onda de la luz. En nuestra retina recibimos la luz en forma de tres estímulos cada uno de los conos tiene una respuesta diferente para cada longitud de onda. El cerebro analiza estas tres señales integrándolas en una sola, esta señal resultante es lo que denominamos color. Un cono responde a las longitudes de onda de la luz que el cerebro luego asocia al rojo, otro a las del verde y otro las de azul. La síntesis de estas tres respuestas generan en nuestro cerebro todos los colores que conocemos.

Por ejemplo el color violeta se generara en el cerebro ante un estímulo de luz azul rojiza, es decir el cono sensible al azul y al rojo reciben la cantidad de luz suficiente para generar el violeta.

El hecho de que la sensación de color se obtenga de este modo, hace que dos objetos observados, radiando un espectro diferente, puedan producir la misma sensación.

3.1.3 Atributos del color

Para cada color que somos capaces de percibir se definen tres atributos que lo definen:

Matiz o tono: es el color en sí mismo, es el atributo que nos permite diferenciar a un color de otro, por lo cual podemos designar cuando un matiz es verde, violeta o anaranjado.

Luminosidad: es la intensidad lumínica de un color (claridad / oscuridad). Es la mayor o menor cercanía al blanco o al negro de un color determinado. A menudo damos el nombre de rojo claro a aquel matiz de rojo cercano al blanco, o de rojo oscuro cuando el rojo se acerca al negro.

Saturación: Es, básicamente, pureza de un color, la concentración de gris que contiene un color en un momento determinado. Cuanto más alto es el porcentaje de gris presente en un color, menor será la saturación o pureza de éste y por tanto se verá como si el color estuviera opaco; en cambio, cuando un color se nos presenta lo más puro posible (con la menor cantidad de gris presente) mayor será su saturación.

Un color viene definido por tanto por sus valores de tono, luminosidad y saturación. Cualquier color se puede identificar con unos valores de estas tres variables. El negro se obtiene cuando no hay brillo (ausencia de luz). Los grises se obtienen cuando la saturación es baja. De los grises el blanco es el más brillante.

3.1.4 Color en monitores

Debido a que el experimento se realizará en una aplicación Web debemos ver como se verá afectado el color por el hecho de estar en un monitor de ordenador.

En las pantallas de computadoras, la sensación de color se produce por la mezcla aditiva de rojo, verde y azul.

La mezcla aditiva consiste en sumar haces de luz de distintas longitudes de ondas (colores), la suma de estos haces forman el nuevo color. El cerebro no distinguirá los componentes de color sumado sino el color que se forma tras la suma. La Fig.2 muestra el ejemplo de muestra aditiva para rojo verde y azul.

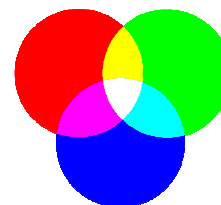


Fig.2

En el monitor hay una serie de puntos minúsculos llamados píxeles. Cada punto de la pantalla es un píxel y cada píxel es, en realidad, un conjunto de tres subpíxeles; uno rojo, uno verde y uno azul, cada uno de los cuales brilla con una determinada intensidad. La intensidad con la que brille cada píxel generará el color en el cerebro.

Este sistema de color recibe el nombre de RGB (red, green y blue; rojo, verde y azul) y define cada color en los tres colores primarios. Cada color primario puede tomar valores del 0 al 255; así el rojo puro tendría las coordenadas RGB (255,0,0).

3.1.5 El color en la Web

Las prestaciones gráficas de los ordenadores de hace unos años limitaban su capacidad de color a 256 colores, debido a que cada color era representado por 2 bits. Además 40 de estos colores se reservaban a la gestión interna del ordenador por lo que la paleta de colores se quedaba reducida a 216 colores.

Las prestaciones del ordenador, el sistema operativo y el navegador que se usa son factores que afectan al color final del sitio Web. Por ello se generó la paleta de colores conocida como “Web safe”, esta paleta abarcaba al principio 216 colores, estos son representados fielmente independientemente del navegador o el sistema operativo.

Con la mejora en los ordenadores la tabla “Web safe” también ha ido aumentando aunque no al mismo ritmo que la reproducción de colores en monitores, en la actualidad se puede disponer de una paleta de colores segura de 4096 colores, lo que es una pequeña cantidad respecto a los millones de colores con los que trabajan las tarjetas gráficas.

3.1.6 Psicología del color.

En la aplicación realizada vamos a ver que preferencia tienen los usuarios respecto a las combinaciones cromáticas y si existe alguna tendencia en función de variables estadísticas. Esto

puede llevar a importantes conclusiones acerca de las emociones subjetivas que producen los colores.

La psicología del color es un campo de estudio que analiza los efectos emocionales que producen los colores y las combinaciones de estos. Es una ciencia aun inmadura, no abundan los estudios en este campo, la fuente mas importante es la “Teoría de los colores” de Goethe.

Conocer la psicología del color puede ayudar a la hora de interpretar las diferentes conclusiones que se establezcan de la aplicación.

Los colores producen sensaciones en el ser humano, aunque la sensación en cada uno es individual y subjetiva si que existen ciertos acuerdos en que los colores producen ciertas emociones comunes en los individuos, que en algunos casos se dan de manera inconsciente, en otros por asociación con la naturaleza o bien por asociaciones culturales.

Los colores son usados como símbolos en la cultura, esto hace que el ser humano asocie los colores a esos símbolos, por ejemplo el rojo se ha usa en Europa para señales de STOP y de peligro, al mismo tiempo se utiliza en el romanticismo como San Valentín. También están las asociaciones con la naturaleza; podemos encontrar el rojo en el fuego, en los labios, en la sangre. Todos estos simbolismos hacen que el rojo produzca una emoción diferente que un color como el azul.

Las emociones que producen los colores son usadas para muchos fines de mercado, el mercado estudia el público al que quiere dirigirse y las sensaciones que quiere producir con su producto y en base a esto elige el color que debe dar a su producto.

Un primer factor a analizar va a ser la temperatura del color. Podemos dividir el espectro de color en colores cálidos y colores fríos, esta división es subjetiva aunque muy extendida. En la fig.3 podemos ver en la izquierda colores cálidos y a la derecha colores fríos, el cerebro realiza estas asociaciones de manera instintiva.



Fig.3

Sabiendo que la cultura y el individuo pueden tener diferentes asociaciones para cada color, vamos a ver ciertas asociaciones psicológicas que de forma más o menos general realiza cada individuo en respuesta a unos colores específicos.

El rojo se utiliza como color excitante, se el asocia el movimiento y el entusiasmo. El naranja es también un color energético sin la pasión del rojo, se asocia a entusiasmo y creatividad. El amarillo es un color muy activo y visible, se asocia a la energía. El verde se asocia a la naturaleza, a la frescura y la esperanza. El azul se asocia a la tranquilidad y el bienestar. El blanco a la pureza y la perfección. El negro se asocia al poder y al éxito y también tiene connotaciones negativas como el mal y la muerte.

3.2 Experimentos en páginas Web

Internet ha sido la gran revolución de los medios de comunicación de los últimos años, esto ha hecho que se haya convertido en una gran herramienta a la hora de realizar experimentos psicológicos o sociológicos.

En estos momentos se realizan numerosos test o experimentos en Internet cuyos participantes son desconocidos para los investigadores. Tampoco se puede llevar un control de observación sobre estos, en comparación con un laboratorio. Las ventajas que Internet aporta a estos experimentos es obvia, la muestra poblacional de estos experimentos es mucho más grande y se consigue en menos tiempo que en otros canales, la posibilidad de conseguir muestras de diversas culturas sin tener que desplazarse, la automatización de la recogida de datos y la oportunidad de estudiar poblaciones específicas de condiciones raras.

Los experimentos en páginas Web también hacen más sencilla la tarea del investigador ya que evita limitaciones horarias y de lugar.

Uno de los problemas que más preocupa en cuanto a este tipo de experimentos es el abandono de los participantes, algo que en un ámbito de laboratorio no sucede puesto que los usuarios se sienten más o menos presionados a finalizar el experimento.

Frick, Bachtiger y Reips (1999) realizaron una de las pocas investigaciones sistemáticas que se han llevado a cabo al respecto, en la que pretendían identificar el papel de distintas variables en el abandono del experimento. Sus resultados mostraron, por un lado, que los incentivos económicos reducen los abandonos pero no influyen sobre las respuestas propiamente dichas y, por otro, que solicitar información personal (como sexo, edad, nacionalidad, dirección de correo electrónico o número de teléfono) antes de comenzar el experimento no solo reduce los abandonos sino que puede permitir obtener datos demográficos mas completos, ya que cuando estos datos se piden al final de la sesión muchas veces quedan en blanco. Otros factores que, de acuerdo con estos autores, pueden influir sobre los abandonos son: el idioma empleado (compararon inglés y alemán), la opinión que se tenga de la institución que está realizando el experimento, el diseño de las páginas y el tiempo que tardan en cargarse.

Otro de los problemas que hay que afrontar es la validez de los datos, ¿Tienen la misma validez un experimento en la Web que en un laboratorio? Para contestar esto, puede considerarse que existen básicamente dos formas: ver si los resultados se corresponden con las teorías de las

cuales se parte y ver si coinciden con los resultados obtenidos con otros métodos considerados tanto válidos como fiables. En este sentido, no solo se ha observado que los resultados provenientes de investigaciones experimentales realizadas a través de la Web se ajustan perfectamente a las expectativas teóricas, sino que trabajos en los que datos obtenidos por ese medio se han comparado con datos de laboratorio (bien recogidos simultáneamente, bien ya publicados) han encontrado una importante coincidencia en los resultados. Y ello independientemente de las características de los participantes de unos u otros estudios y de las diferentes condiciones ambientales.

En el caso de este experimento, Internet nos aporta todas sus ventajas. La realización del experimento se puede hacer de forma más sencilla e intuitiva desde un monitor de ordenador, además al ser un estudio sobre combinaciones cromáticas en páginas Web, que mejor que mostrar estas páginas en su medio natural. Para evitar el abandono y para mejorar las posibles conclusiones el usuario de la aplicación dará una serie de datos con fines estadísticos.

3.3 Ordenación

Como se especifica en el objetivo cada usuario debe ordenar una lista de imágenes mediante un método de comparación entre dos imágenes. Las imágenes a ordenar deben ser variadas y su número significativo, aun así no se pretende que un solo usuario ordene una gran lista de imágenes, puesto que esto puede suponer un cansancio y hartazgo en el participante que se traduzca en falta de interés para el experimento y por tanto unos resultados no válidos.

El problema a resolver es elegir un método de ordenación eficiente y un número de imágenes que no supongan demasiadas comparaciones para el usuario de forma que le resulte un experimento sencillo y breve. Una vez elegido el número de imágenes para cada usuario se elegirá un número total de imágenes, de forma que unos pocos usuarios puedan cooperativamente ordenar la lista total de imágenes.

Existe una lista importante de algoritmos de ordenación. Un algoritmo es un conjunto preescrito de instrucciones que realizan una actividad, en el caso de los algoritmos de ordenación dicha actividad consiste en tener como entrada una lista desordenada y como salida la misma lista ordenada. Algunos algoritmos de ordenación son el Bubblesort, el Cocktail sort, el Insertion sort o el mergesort.

Uno de los factores para elegir un algoritmo de ordenación es la complejidad computacional, la complejidad indica el número de pasos necesario para resolver el algoritmo. En este caso elegimos el algoritmo mergesort ya que su complejidad es logarítmica, lo que indica que necesita menos pasos para ser llevados que otros algoritmos de complejidad exponencial o lineal.

3.3.1 Algoritmo mergesort

Al usuario se le presentan una cantidad de imágenes de la siguiente forma, primero se le presentan dos imágenes y elige la que prefiere de las dos, luego se le presentan otras dos y realiza la misma acción y así sucesivamente.

Para establecer un orden claro de preferencia de cada usuario entre todas las imágenes presentadas, minimizando la cantidad de pares de imágenes y la repetición de estas, se emplea el algoritmo mergesort.

El algoritmo de ordenación mergesort se basa en la técnica del divide y vencerás, tiene una complejidad de:

$$\Theta = n \log_2(n)$$

Este algoritmo fue desarrollado en 1945 por John Von Neumann, se emplea para el ordenamiento de arrays y funciona de la siguiente manera:

- . Si la lista o array es de tamaño 1 o 0 ya está ordenada.
- . Dividir la lista en dos mitades.
- . Ordenar cada sublista por el método mergesort.
- . Mezclar cada sublista en una sola lista ordenada.

El algoritmo mergesort es recursivo, este es el pseudocódigo que explicaría el algoritmo:

```

Mergesort(A[1..n])
IF n es pequeño
RETURN Inserción(A)
ELSE
  crear A1 y A2 subarreglos de A
  B1 ::= Mergesort(A1)
  B2 ::= Mergesort(A2)
  Mezcla(B1, B2, B)
  RETURN B
ENDIF
  
```

Es decir partimos la lista en dos sublistas, cada sublista la partimos en dos, así sucesivamente hasta que el tamaño de la sublista sea 1 o 0, entonces ordenamos los elementos de la sublista por comparación, cuando dos sublistas ya están ordenadas se pueden unir en una sublista hasta obtener la lista ordenada.

En la Fig.4 se muestra un ejemplo de ordenación de un array de 7 elementos mediante el algoritmo mergesort.

Como podemos apreciar en la Fig.4 se subdivide el array hasta obtener elementos simples y es entonces donde se realiza la comparación, en el caso de este proyecto este será el momento en el que el usuario elija entre una imagen u otra, estableciendo el usuario el ordenamiento de las fotos según su preferencia.

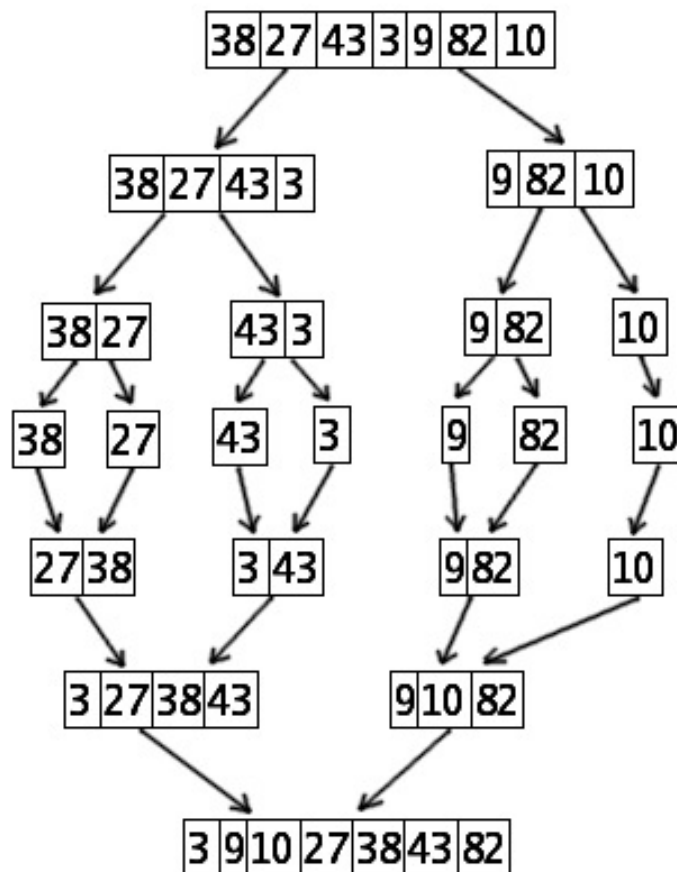


Fig.4

Veamos como el algoritmo merge sort ordenaría de mayor a menor una lista de 8 números desordenada

Lista desordenada= [2, 5, 7, 4, 1, 3, 8, 6]

1- Ordenamiento por elementos simples:

Primero se toma cada valor simple como elemento ordenado y se compara uno a uno, dejando la lista ordenada de dos en dos. La primera comparación es de los dos primeros elementos, como el 5 es mayor que el 2 el 5 se colocaría primero en la lista ordenada y seguido el 2.

Listas ordenadas=> [2] [5] [7] [4] [1] [3] [8] [6]

Comparación

¿[2] ó [5]? $5 > 2$

¿[7] ó [4]? $7 > 4$

...

Lista ordenada \Rightarrow [5, 2, 7, 4, 3, 1, 8, 6]

2- Ordenamiento por pares

Tras el primer paso nos encontramos con cuatro sublistas de dos elementos ordenadas, comparamos cada sublista con la siguiente de la siguiente manera: comparamos el primer elemento de cada sublista; el que sea mayor ocupará el primer lugar de la nueva lista ordenada, después se compara el elemento siguiente al que ha resultado ser el mayor con el elemento que antes ha sido menor; si el antiguo menor sigue siendo menor a la nueva comparación entonces la lista ya estará ordenada, si no se compara con el siguiente a él mismo.

Listas ordenadas \Rightarrow [5, 2] [7, 4] [3, 1][8, 6]

Comparación

[5, 2] y [7, 4]

¿5 ó 7? $7 > 5$

¿5 ó 4? $5 > 4$

¿2 ó 4? $4 > 2$

[3, 1] y [8, 6]

¿3 ó 8? $8 > 3$

¿3 ó 6? $6 > 3$

Lista ordenada \Rightarrow [7, 5, 4, 2, 8, 6, 3, 1]

3- Ordenamiento de cuartetos

Después de la anterior ordenación, nos encontramos con dos sublistas ya ordenadas de 4 elementos, la comparación se realizará igual que en el anterior paso, primero los dos primeros elementos de cada lista y en la siguiente comparación se compara el siguiente al que ha resultado mayor de los dos.

Listas ordenadas \Rightarrow [7, 5, 4, 2] y [8, 6, 3, 1]

Comparación

[7, 5, 4, 2] y [8, 6, 3, 1]

¿7 ó 8? $8 > 7$

¿7 ó 6? $7 > 6$

¿5 ó 6? $6 > 5$

¿5 ó 3? $5 > 3$

¿4 ó 3? $4 > 3$

¿3 ó 2? $3 > 2$

¿2 ó 1? $2 > 1$

Lista final ordenada= [8, 7, 6, 5, 4, 3, 2, 1]

En este ejemplo cada pregunta es una comparación del usuario y ha habido un total de 16 preguntas, lo que supondría comparar 16 números para ordenar la lista de 8 números.

En el caso de las imágenes se podría decir que al clicar sobre una imagen el usuario esta escogiendo a esa imagen como mayor que la otra, con lo que al final de la comparación habría conseguido ordenar la lista entera de imágenes de mayor a menor como si de números se tratara.

La complejidad del algoritmo mergesort viene definida por la función $\Theta = n \log_2(n)$, la función Θ es la cota superior, es decir los tiempos de ejecución para el peor caso posible, siendo n el número de elementos de la lista a ordenar

Elección del número de imágenes

Para elegir el número de imágenes que presentaremos al usuario debemos aclarar antes el número de iteraciones que se quiere presentar al usuario, el objetivo no es conseguir el hartazgo del usuario ante la continua sucesión de pares de imágenes por eso establecemos un número de iteraciones que oscile entre 15 y 20.

Para calcular las imágenes necesarias usamos la función Θ del algoritmo mergesort. Esta función nos dará las máximas iteraciones para el ordenamiento de una lista de “n” elementos y obtenemos los siguientes datos:

Para $n=10$	$\Theta=33.18$;
Para $n=9$	$\Theta=28.52$;
Para $n=8$	$\Theta=24$;
Para $n=7$	$\Theta=19.65$;

A la vista de estos resultados parece que el número idóneo de imágenes a presentar será 7 o 8.

Como Θ representa el máximo número de iteraciones puede que este dato no sea suficiente para elegir el número imágenes a presentar, por ello vamos a calcular la media de iteraciones para estos valores.

Para ello se ha implementado el algoritmo en el archivo “merge.php”, en este PHP se ordena un array de “n” elementos de forma recursiva, incluyendo un contador que se auto incrementa en el momento de comparación valor a valor (iteraciones mostradas al usuario). Se

ejecuta el “php” cambiando el valor de n y se obtiene un número promedio de iteraciones basado en 15 pruebas por cada valor de n.

Para n=10	Promedio= 22.8;
Para n=9	Promedio= 19.4;
Para n=8	Promedio= 14.9;
Para n=7	Promedio= 12.6;

Podemos observar que el promedio de iteraciones para 7 imágenes es demasiado bajo para el valor que esperamos.

Para 8 imágenes obtenemos un promedio de iteraciones que ronda el rango inferior que nos hemos marcado como óptimo y el máximo está algo por encima del rango superior óptimo, por tanto cada usuario ordenará una lista de 8 imágenes.

Elegimos 64 imágenes de páginas Web de tal forma que 8 usuarios puedan ordenar de manera cooperativa las 64 imágenes.

Con estas conclusiones obtenemos que **cada usuario ordena una lista de 8 imágenes de tal forma que cada 8 usuarios ordenen el total de 64 imágenes.**

4 MATERIALES Y MÉTODO

4.1 Herramientas

Al realizarse el experimento en una aplicación Web muchas de las herramientas van orientadas al diseño de páginas Web dinámicas, en este caso la aplicación Web está desarrollada en HTML y Php con SQL. Los análisis estadísticos también han sido realizados con software dirigido a estadística.

4.1.1 HTML

HTML (HyperText Markup Language, en español: “Lenguaje de Marcado de Hipertexto”) es imprescindible para la creación de cualquier sitio Web. No es un lenguaje de programación sino un lenguaje de marcado, se basa en etiquetas que definen la estructura y el contenido del texto en páginas Web, también complementa el texto con otros elementos como imágenes, formularios, tablas, hiperenlaces...

HTML está definido por estándares que permiten la compatibilidad de navegadores y facilitan el diseño Web.

Un ejemplo del lenguaje de HTML es el siguiente:

```
<HEAD>  
<TITLE>Aquí va el titulo de la página </TITLE>  
</HEAD>
```

En este caso la etiqueta “<HEAD>” sirve de cabecera, dentro de ella se introduce el título, engloba una serie de sentencias a modo de cabecera como son por ejemplo, el título de la página, los enlaces a archivos externos como hojas de estilo css y otras. La etiqueta “<TITLE>” da título a la página, en este caso en la pestaña del navegador donde se reproduce este código estaría escrito “Aquí va el titulo de la página”.

4.1.2 PHP

PHP es un lenguaje de programación orientado originalmente a la creación de páginas Web dinámicas.

PHP nos permite incrustar sus pequeños fragmentos de código o scripts dentro de la página HTML y realizar determinadas acciones de una forma fácil y eficaz, combinándolo con HTML. Es decir, con PHP escribimos scripts dentro del código HTML

Por otra parte PHP ofrece un sinfín de funciones para la explotación de bases de datos de una manera llana y sin complicaciones.

Todas las conexiones con bases de datos y las operaciones con los datos de los usuarios y las imágenes están implementadas en PHP.

Para introducir código PHP dentro de HTML se utiliza la siguiente etiqueta de script:

Comienzo del código PHP: `<?php`

Fin del código PHP: `;>`

4.1.3 SQL

El lenguaje SQL es el que se utiliza para interactuar con una base de datos.

Es un lenguaje normalizado que nos permite trabajar con cualquier tipo de lenguaje (en este caso utilizaremos PHP) en combinación con cualquier tipo de base de datos (en este caso MySQL).

Para realizar cualquier consulta a la base de datos utilizamos código de PHP insertado en el HTML, este código a su vez contendrá código SQL, este código SQL es el que modificará la base de datos.

Ejemplo de consulta desde php:

```
$consulta3=      "SELECT *
FROM `Infousuarios`
ORDER BY `id` DESC
LIMIT 1
";
$resultado= mysql_query($consulta3 , $conexion) or die(mysql_error());
```

En este caso el texto en rojo sería lenguaje SQL, estaría realizando una consulta de tipo “select”, que sirve para seleccionar un registro de una tabla.

4.1.4 Servidor y Base de datos

El servidor donde se aloja la aplicación Web lo proporciona el servicio informático de la UPNA, es un servidor de HTTP Apache la versión 2.0.

Al servidor se le ha dado un nombre DNS único (Domain Name System; en español: sistema de nombre de dominio), que agrupa la IP interna para conexiones dentro de la UPNA y resuelve la IP pública.

El acceso al servidor y por tanto a la página Web se realiza desde el siguiente nombre DNS:

`http:// colorlab.unavarra.es:8000`

El acceso al FTP del servidor se realiza desde la IP privada del servidor, que es la 172.18.197.107 ; La base de datos proporcionada es MySQL.

4.1.5 phpMyAdmin

phpMyAdmin es una herramienta que permite una sencilla administración y manejo de las bases de datos de MySQL a través de una interfaz Web.

El primer paso es instalar y configurar el phpMyAdmin para el servidor y la base de datos.

Después se crea una cuenta con privilegios, desde esta cuenta se pueden crear y eliminar bases de datos, crear, eliminar y alterar tablas, borrar, editar y añadir campos, ejecutar cualquier sentencia SQL, administrar claves en campos y exportar datos en varios formatos.

La interfaz está disponible en varios idiomas lo que simplifica más su uso.

4.1.6 Software

Dreamweaver

Para diseñar la aplicación Web y la programación en PHP se ha utilizado el Adobe Dreamweaver CS4 versión 10.0.

El Dreamweaver es una herramienta-estudio enfocado a la construcción y edición de sitios Web, es un editor WYSIWYG (What You See Is What You Get, en español: “lo que ves es lo que obtienes”); esto implica que puedes editar páginas de forma visual y este se encarga de ir escribiendo el código. También permite el acto contrario, escribir código e ir viendo los resultados de forma visual. Es importante compaginar las dos técnicas.

También permite ver el código PHP de una forma clara y diferenciada, que ayuda a la hora de programar.

Aunque estas herramientas son de gran utilidad ya que simplifican mucho el trabajo del diseño de páginas Web siempre es necesario conocer ciertas nociones básicas del lenguaje HTML para poder desarrollar el sitio de forma simple y eficaz.

FileZilla

FileZilla es una aplicación para la transferencia de archivos por FTP. Es una aplicación gratuita y de código abierto

Permite transferir archivos desde una computadora local, hacia uno o más servidores FTP (y viceversa) de forma sencilla.

También permite una vista sencilla de archivo y carpeta, situada debajo del registro de mensajes, proporciona una interfaz gráfica para FTP. Los usuarios pueden navegar por las carpetas y ver y alterar sus contenidos tanto en la máquina local como en la remota, utilizando una interfaz de tipo árbol de exploración. Los usuarios pueden arrastrar y soltar archivos entre los ordenadores local y remoto.

Se ha utilizado FileZilla para subir todo el contenido de la Web y para actualizarlo.

SPSS

Statistical Package for the Social Sciences (SPSS) es un programa estadístico informático. Es capaz de trabajar con grandes bases de datos y permite un manejo sencillo de los datos y de los estadísticos.

4.2 Conjunto de imágenes

Las imágenes corresponden a 64 páginas Web con diferentes combinaciones cromáticas. Estas páginas han sido seleccionadas basándose únicamente en su combinación cromática, su contenido es irrelevante.

A continuación en la Fig.5 se muestran las 64 imágenes elegidas, cada una de ellas va asociada a un número, durante toda la aplicación y en el análisis de los resultados todas las imágenes se referenciarán a estos números para hacer más sencillo su manejo.

En el anexo podemos encontrar otros criterios utilizados para elegir las imágenes.

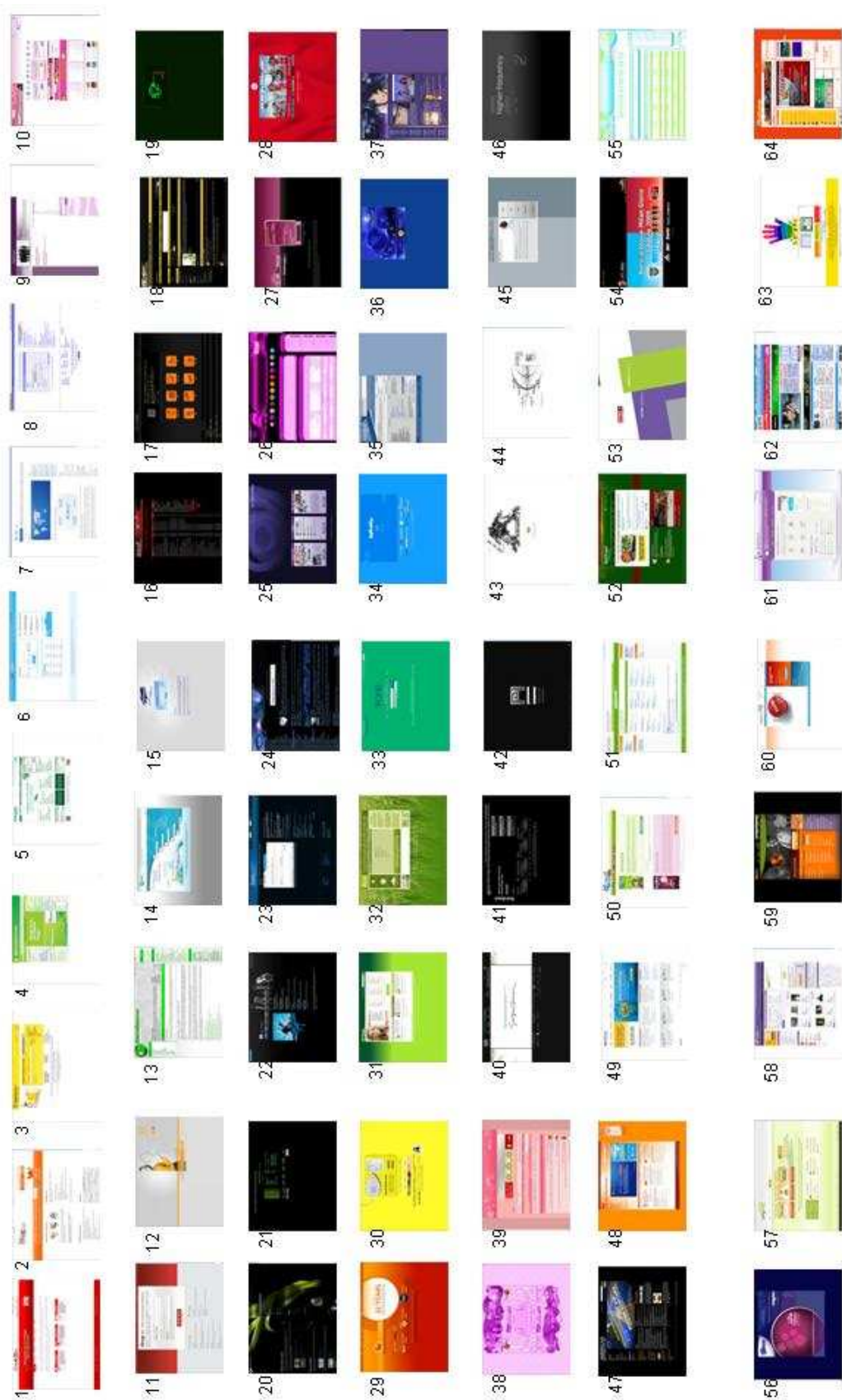


Fig.5

4.2.1 Clasificación según criterios

Para el posterior análisis de resultado se ha dividido el conjunto de imágenes según los criterios de estudio con esto se llega a tres clasificaciones distintas: Según su nivel de luminosidad de color (claro, medio y oscuro), según su fondo (blanco, gris, negro...) y según el tono predominante(rojo,azul...).

Las imágenes han sido clasificadas de manera visual.

Criterio según Luminosidad

El primer criterio de clasificación es clasificar las imágenes según su luminosidad, para ello dividimos las imágenes en imágenes claras, medias y oscuras, esto se realiza de manera visual.

Las claras las componen las imágenes con fondo blanco, de colores muy luminosos y la de fondo gris claro. Las imágenes de luminosidad media están formadas por monocromáticas, por imágenes de tonos medios e imágenes de fondo de tonos o grises medio. Las oscuras están compuestas por imágenes de fondo negro o tonos oscuros y monocromáticas de tonos oscuro y gris oscuro.

En la Fig. 6 se muestra las 64 imágenes divididas en los tres criterios.

Clasificación según el fondo

Clasificamos las imágenes según su fondo, tenemos así cuatro divisiones, de fondo blanco, de fondo negro, de fondo gris y de fondo cromático. Hay dos imágenes de toda la muestra que no tienen un fondo concreto, estas dos imágenes no formarán parte de este criterio.

Como se aprecia en la Fig.7 las dos primeras líneas de imágenes corresponden a las de fondo blanco, las dos siguientes a las de fondo negro, la línea posterior a las páginas de fondo gris y las dos últimas líneas son imágenes con fondo cromático.

Clasificación según el tono

El último criterio de clasificación es según el tono, para cada tono se tomarán las imágenes en los cuales dicho tono sea predominante. El blanco y el negro no son tonos por tanto la combinación de estos con un tono se tomarán como parte del tono. Las imágenes acromáticas (blanco, negro y gris) se tomarán como una división aparte.

No todas las imágenes tienen un tono predominante, para esta clasificación dichas imágenes no se han utilizado.

En la Fig. 8 se encuentran las imágenes utilizadas divididas según su tono predominante.

Claros



Medias



Oscuras



Fig.6

Fondo Blanco



Fondo Negro



Fondo Gris



Fondo Color



Fig.7



Fig.8

4.3 Métodos estadísticos de análisis

4.3.1 Introducción

El primer paso para ver que estadísticas necesitamos emplear para extraer conclusiones es analizar el tipo de datos que se manejan en el experimento.

A cada usuario se le presenta una lista de 8 imágenes y este las ordena mediante comparación de dos en dos. Cada 8 usuarios se ordenará la lista completa de 64 imágenes.

Con esta ordenación se puede establecer el orden en el que el usuario prefiere su lista de imágenes. No especifica el grado de preferencia para cada foto ni la proporción de preferencia respecto a las otras fotos; es decir el experimento no dice cuanto le gusta al usuario una imagen, ni cuanto la prefiere respecto a otra, simplemente proporciona el orden en que prefiere la lista de ocho imágenes.

La imagen mejor valorada de cada lista recibe 8 puntos, la siguiente 7 puntos y así hasta llegar a la peor valorada que recibe 1 punto. Esta puntuación impuesta representa el orden, no una valoración cuantitativa de las diferencias en la preferencia de las imágenes; no se puede determinar si las preferencias en cada imagen son equidistantes, incluso el experimento no permite la opción de equiparar la preferencia entres dos imágenes.

Estas características de los tipos de datos del experimento, que se denominan datos ordinales hace que la estadística no se ajuste a los llamados criterios paramétricos y que haya que usar por tanto una estadística no paramétrica para el análisis de los datos del experimento.

Estadística no paramétrica

La estadística no paramétrica es una rama de la estadística que estudia las pruebas y modelos estadísticos cuya distribución subyacente no se ajusta a los llamados criterios paramétricos. Su distribución no puede ser definida previamente, puesto que son los datos observados los que la determinan. La utilización de estos métodos se hace recomendable cuando no se puede asumir que los datos se ajusten a una distribución conocida, cuando el nivel de medida empleado no sea, como mínimo, de intervalo o en una escala de razón.

En este caso las medidas no siguen estos criterios paramétricos y se hace necesario la estadística no paramétrica para obtener conclusiones con las medidas obtenidas.

Existen una gran cantidad de pruebas no paramétricas, para este experimento utilizaremos tres de ellas: la prueba de Friedman, la prueba de Wilcoxon y la prueba de Mann-Whitney.

4.3.2 Prueba de Friedman

La prueba de Friedman es el equivalente al análisis de varianza de Fisher para dos factores en su versión no paramétrica.

La prueba de Friedman permite contrastar si existen diferencias entre las poblaciones de las que fueron extraídas k muestras, cuando éstas corresponden a datos relacionados. Éstos han de estar medidos al menos en una escala ordinal.

En nuestro caso las k muestras pueden ser por ejemplo las mujeres de la muestra que han comparado las imágenes rojas y amarillas, y datos relacionados se refiere a que toda la muestra ha comparado los mismos datos; Por ejemplo si quisiéramos comparar la preferencia en imágenes rojas entre hombres y mujeres no podríamos realizar esta prueba, puesto que las muestras a estudiar no tienen relación, son independientes.

La prueba de Friedman se realiza de la siguiente manera:

1. Se plantea la hipótesis:

Hipótesis nula: No existen diferencias entre los grupos.

Hipótesis alternativa: Hay diferencias entre los grupos.

2. Estadístico de contraste

El estadístico de contraste utilizado es χ_r^2 (Chi-cuadrado) que se determina a partir de los siguientes pasos:

- a) En una tabla; se colocan todas las puntuaciones de forma que las filas son los sujetos y las columnas son las muestras.
- b) Se ordenan los sujetos en cada fila.
- c) Se calculan las sumas de rangos R_j correspondientes a cada columna.
- d) Se obtiene el estadístico de Friedman.

$$\chi_r^2 = \frac{12}{Nk(k+1)} \sum_{j=1}^k (R_j)^2 - 3N(k+1)$$

3. Consultamos la tabla correspondiente para obtener los valores críticos. En ella se recogen las probabilidades asociadas a valores tan grandes como χ_r^2 . Si el valor observado es igual o mayor que el valor crítico para un nivel de significación determinado, podemos rechazar la hipótesis nula.

4.3.3 Prueba de Wilcoxon

Esta prueba se utiliza cuando queremos averiguar si dos tratamientos o niveles de la variable- independiente se diferencian entre sí (prueba bilateral) o uno es mejor que otro (prueba

unilateral). En este caso, los sujetos han de ser los mismos sujetos o estar apareados en los dos tratamientos.

La única condición que debemos tener en cuenta para aplicar, esta prueba es que tenemos que conocer cuál de las dos puntuaciones de cada par es mayor y que podamos ordenar las diferencias sin tener en cuenta su signo (valor absoluto).

Esta prueba se puede utilizar por ejemplo cuando queramos comparar la valoración de dos grupos independientes como hombres y mujeres para una misma variable como por ejemplo la preferencia por las imágenes rojas, esta prueba nos dice si existen diferencias significativas entre los dos grupos. La prueba se realiza de la siguiente manera:

1. Primero se plantea la hipótesis:

Hipótesis nula: No existen diferencias entre los dos grupos o tratamientos.

Hipótesis alternativa: Hay diferencias entre los dos grupos.

2. Estadístico de contraste.

El estadístico de contraste es la W de Wilcoxon. Para su cálculo, debemos seguir los siguientes pasos:

a) Para cada par igualado, se calcula diferencia entre las dos puntuaciones.

b) Se ordenan de menor a mayor las diferencias, sin tener en cuenta el signo.

No se incluyen las diferencias con valor igual a cero.

c) A cada diferencia se le asigna un rango a partir de la clasificación, y se coloca el signo correspondiente.

d) Se calcula W como el valor más pequeño posible al sumar todos los rangos positivos (W_+) y todos los negativos (W_-).

En la práctica, no es necesario realizar todos estos cálculos, puesto que ya existen tablas construidas al efecto.

3. Comparación con el valor teórico.

Una vez obtenido el valor de W se compara con el correspondiente valor crítico, correspondiente al número de diferencias no nulas. En el caso de que sea inferior a él, podremos rechazar la hipótesis nula y aceptar la hipótesis alternativa.

Si estamos ante un contraste unilateral, nos interesa conocer si las puntuaciones de una muestra (X) son superiores a las de la otra (Y), quedando formuladas las correspondientes hipótesis de la siguiente forma:

Hipótesis nula: Las puntuaciones X son iguales o inferiores a las puntuaciones Y.

Hipótesis alternativa: Las puntuaciones X son superiores a las puntuaciones Y.

Si la hipótesis nula es cierta, esperamos que las diferencias $Y-X$ sean de signo positivo, y que, por tanto, al sumar todos los rangos positivos obtendríamos un valor elevado. Para rechazar la hipótesis nula, tendríamos que encontrar valores de W_+ que se alejen de los esperados de acuerdo con la hipótesis nula, o sea, valores pequeños de W_+ .

Si la muestra es mayor que 20, es difícil encontrar tablas que recojan los valores críticos, ya que la forma de obtenerlos se complica en demasía. No obstante, es posible calcularlos partiendo del supuesto de que las muestras grandes siguen una distribución normal, con los siguientes parámetros media y desviación típica:

$$\mu_w = \frac{N(N+1)}{4} \quad \sigma_w = \sqrt{\frac{N(N+1)(2N+1)}{24}}$$

Por lo tanto, el estadístico W se transforma en una puntuación típica y sigue una distribución normal con media de 0 y desviación típica de 1, y podremos obtener la probabilidad asociada al mismo a partir de la tabla de valores de la distribución normal.

4.3.4 Prueba de Mann-Whitney

La prueba de Mann-Whitney se emplea en aquellos casos en los que deseamos contrastar si existen diferencias entre las poblaciones de donde se extrajeron dos muestras, que han de ser aleatorias e independientes. La utilidad de esta prueba es la misma que la de la prueba t de Student, pero no parte de supuestos y puede ser aplicada a datos medidos en escala ordinal.

Esta prueba se puede utilizar por ejemplo para comparar una misma variable como por ejemplo el tono rojo y la puntuación de dos poblaciones diferentes e independientes, como mujeres y hombres y comprobar si existen diferencias significantes.

El procedimiento es el siguiente:

1. Primero se plantea la hipótesis:

Hipótesis nula: No existen diferencias entre los dos grupos.

Hipótesis alternativa: Hay diferencias entre los dos grupos.

2. Estadístico de contraste:

En este caso, el estadístico a emplear se denomina U de Mann-Whitney, que se calcula siguiendo estos pasos:

a) Se procede a ordenar las puntuaciones de las dos muestras como si fueran una sola.

b) A cada una de ellas se le asigna un rango.

c) Se calcula el estadístico T, a partir de la suma de los rangos de la muestra de menor tamaño.

d) Teniendo T, se calcula U:

$$U = n_1 n_2 + \frac{n_1(n_1+1)}{2} - T$$

Donde n_1 es el número de sujetos de la muestra de menor tamaño, y n_2 el de la muestra menor.

3. El valor observado es comparado con valores tabulados.

Para muestras grandes el SPSS ofrece una tipificación del estadístico aproximándolo a una distribución normal $N(0,1)$.

5 APLICACIÓN

5.1 Descripción General

El usuario entra en la página de portada, en la que puede elegir el idioma en el que quiere realizar el test. Se le muestran las instrucciones del test (*vease Instrucciones*) y pasa a rellenar un formulario con unos pocos datos con fines únicamente estadísticos.

Una vez rellenado el formulario se elige una lista de 8 fotografías para el usuario, en el archivo “listas.txt” se encuentran las distintas combinaciones aleatorias de 8 números, cada número va asociado a una imagen.

Esta elección se determina mediante el valor “m”, este valor hace de puntero para la elección de la lista de 8 imágenes en el archivo “listas.txt”, se incrementa en uno cada vez que el usuario finaliza la aplicación. Así el siguiente usuario cogerá la siguiente lista de fotos del archivo “listas.txt”.

De la lista elegida se muestra al usuario dos imágenes, el usuario elige la que mas le gusta, así sucesivamente hasta ordenar la lista de 8 imágenes siguiendo el algoritmo mergesort.

Cada imagen recibe una valoración del 1 al 8 según el orden de preferencia siendo el 8 la preferida por el usuario y el 1 la menos valorada de todas.

Para cada usuario se guarda en la base de datos el número asociado a las fotos que se le han mostrado con sus respectivas puntuaciones

Se guarda también el tiempo total del test, el tiempo medio de decisión entre dos imágenes, el tiempo mínimo de decisión y el tiempo máximo, para cada usuario del test.

En una lista diferente se guarda las veces que ha sido mostrada una foto y su puntuación total, de estos dos datos se puede obtener la media de puntos por cada foto.

Como retroalimentación se presenta al usuario su lista de imágenes según el orden resultante de su elección; también se le presenta la misma lista de fotos ordenadas según la valoración media de todos los usuarios.

Si el usuario decidiera repetir el test, se le mandaría de nuevo al formulario, dándole una nueva lista de 8 imágenes.

5.2 Esquema aplicación Web

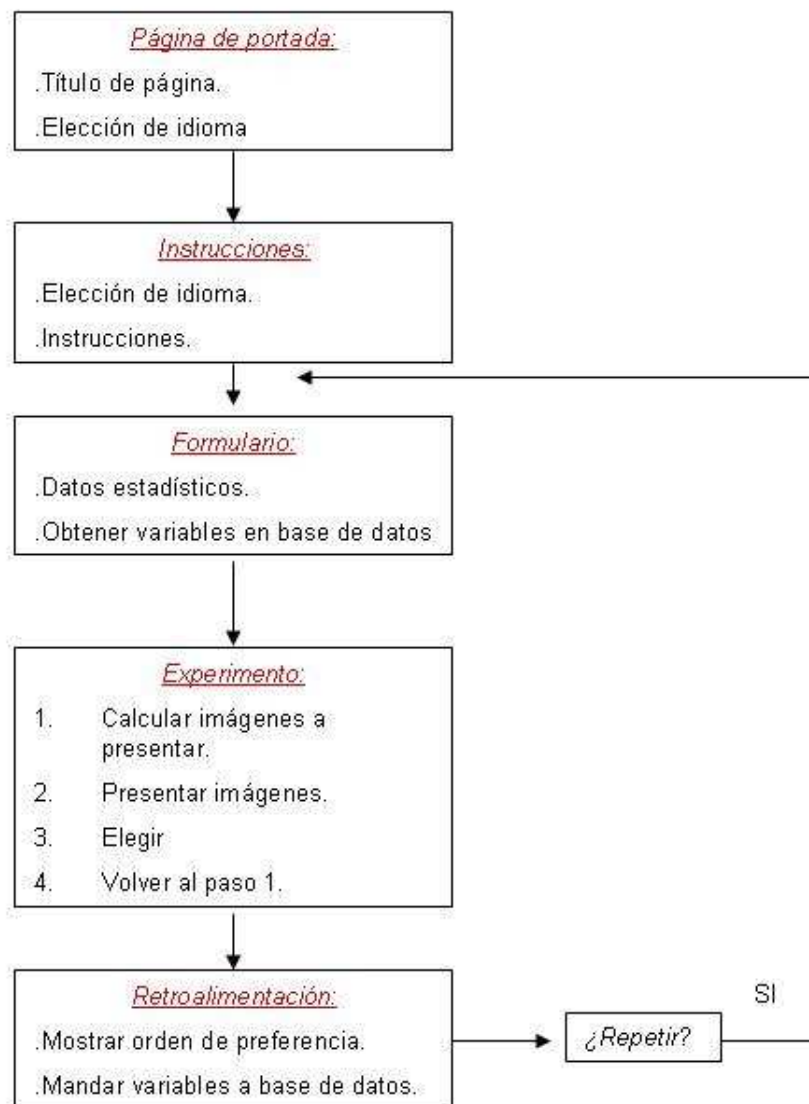
Para explicar el funcionamiento de la aplicación Web dividiremos el contenido en tres grandes apartados.

En el primer apartado se muestran el aspecto visual de las páginas que se presentan al usuario para realizar el test, en el segundo apartado veremos de que forma se almacenan los datos en la aplicación y en el último apartado veremos como funciona la aplicación, analizando sus diferentes rutinas implementadas.

Al usuario se le presentan un total de cinco páginas diferentes, las tres primeras son a modo introductorio al experimento y para que este complete el formulario con sus datos, la siguiente página es donde se realiza el experimento, esta página se ejecuta las veces necesarias para ordenar la lista de imágenes desordenadas, por último se muestra la retroalimentación. Durante todo este proceso la aplicación mientras muestra las diferentes pantallas al usuario va almacenando datos y corriendo rutinas para el funcionamiento de la misma.

En la Fig.9 se muestra el esquema de funcionamiento básico de la aplicación Web, los detalles se concretan en apartados posteriores.

Fig.9



5.3 Aspecto visual de la aplicación

Como antes se ha dicho para realizar el experimento al completo se pasa por un total de cinco páginas distintas, en este apartado vamos a ver la apariencia gráfica de estas páginas y explicar brevemente el contenido de las mismas.

La aplicación está en diferentes idiomas pero para evitar la repetición mostraremos la aplicación solo en castellano, el diseño en los otros idiomas es idéntico.

5.3.1 Índice

Al acceder a la URL de la aplicación la página de portada que se muestra es la de la Fig.10, en esta se muestra una breve introducción al grupo de COLORLAB y la opción de elegir uno de los dos experimentos de dicha página, también está la opción de cambiar de idioma para esta página, clickando en uno de los 3 idiomas que hay en la parte superior del texto.

Para comenzar con el experimento que ocupa este proyecto hay que clicar en empezar o en el propio título del proyecto.

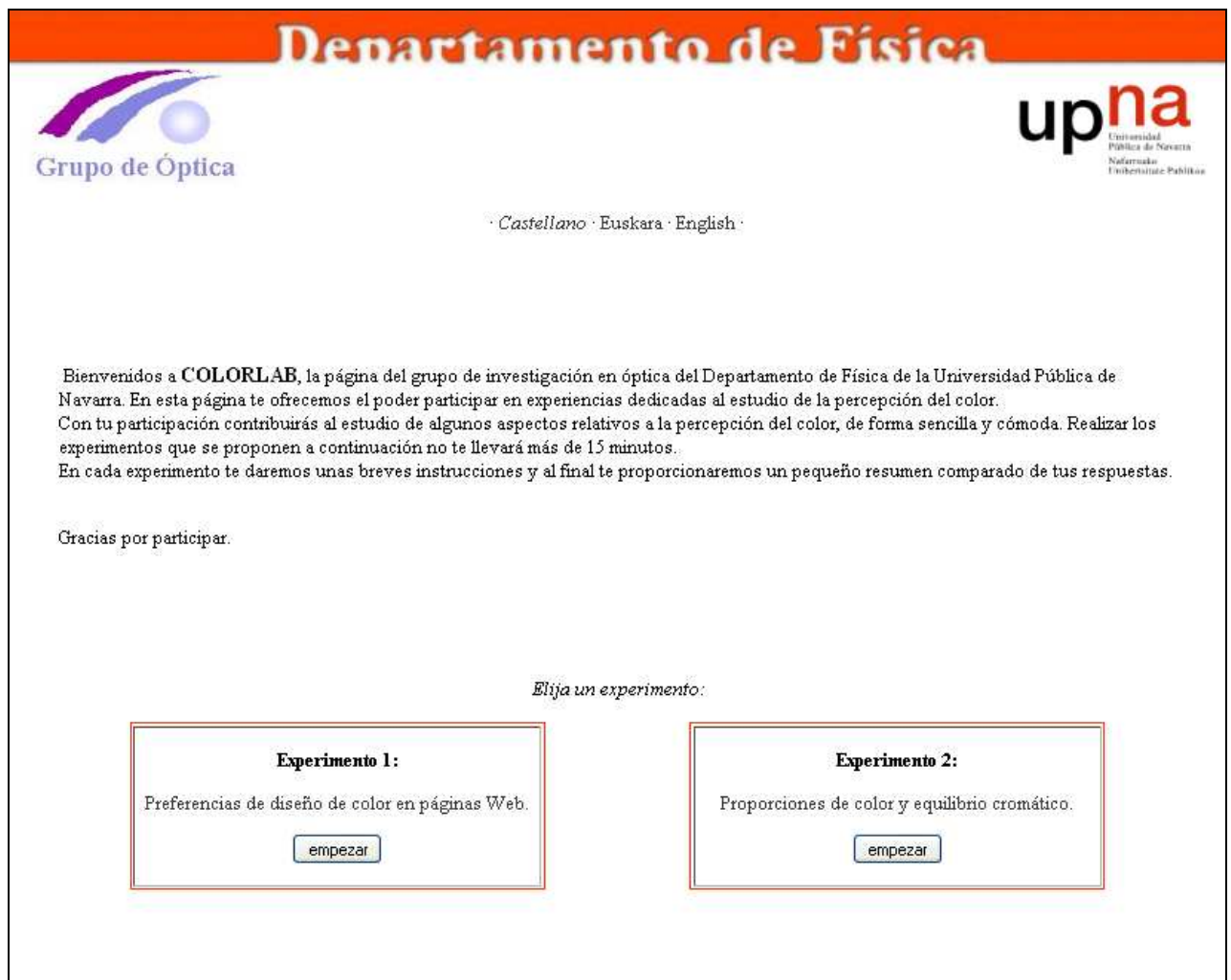


Fig.10

5.3.2 Portada

La página de portada es la introductoria al experimento que ocupa este proyecto, en ella se encuentra el título del proyecto y la posibilidad de cambiar de idioma, esta página sirve en el caso de que se quiera presentar a alguien solo este proyecto, en dicho caso se puede acceder directamente al proyecto introduciendo la URL correspondiente a dicha página: colorlab.unavarra.es:8000/Pr/limpio/portada.php.

En la Fig.11 podemos ver el aspecto visual de dicha página.

Fig.11



5.3.3 Instrucciones

En esta página se encuentran las instrucciones del experimento, como en los anteriores se puede cambiar de idioma en la misma página, en enlace para continuar a la siguiente página esta en la última palabra de las instrucciones: “aquí”.

En la Fig. 12 podemos ver el aspecto visual de esta página.

Seleccione su idioma Castellano

Instrucciones.

En primer lugar, gracias por participar. La realización de este experimento es anónima y su duración aproximada de 10 minutos. Al final les proporcionaremos información sobre sus resultados.

Para su correcta ejecución, le rogamos que lea atentamente las siguientes explicaciones e instrucciones.

Objetivos del experimento.

Conseguir que una página Web resulte visualmente atractiva es sin duda un objetivo importante en el momento de su diseño. Entre otros factores, la disposición de la información, el diseño de los menús de navegación o el uso del color influyen en el aspecto final y en el atractivo de una página Web.

En este experimento nos centraremos en la influencia del color. En concreto nuestro objetivo es determinar cuáles son las preferencias de los usuarios en cuanto al uso del color y de las combinaciones de colores en páginas Web.

Desarrollo del experimento.

La realización del experimento es muy sencilla. En primer lugar le pediremos algunos datos estadísticos (sexo, edad, etc) y seguidamente comenzaremos con el experimento.

Para conocer sus preferencias, el programa le mostrará dos imágenes, una al lado de la otra, como se muestra en la figura. Se trata de imágenes correspondientes a páginas Web reales.

Imagen 1

Imagen 2

Haga CLICK con el ratón sobre la imagen cuyo
diseño de color le parece mas atractivo

Fíjese en las imágenes, centre su atención únicamente en el color y su distribución. No se deje influenciar por otros aspectos que no sean únicamente el color.

Dado que las imágenes se han obtenido de páginas Web reales es posible que conozca alguna de ellas. No se deje influenciar por este hecho. No se fije tampoco en los contenidos de la página. Concentre su atención en el uso del color que se hace en cada una de ellas.

Observe cada pareja de imágenes unos segundos, no emplee demasiado tiempo. Una vez que haya decidido cuál es la imagen que más le gusta, pulse sobre ella con el ratón.

Según vaya contestando, el programa le mostrará nuevas parejas de imágenes. Observará que las distintas imágenes se van repitiendo, aunque nunca le mostraremos la misma pareja dos veces. Esto nos permitirá conocer perfectamente sus preferencias.

Si ha leído y comprendido estas instrucciones, puede pasar a realizar el experimento pulsando [aquí](#)

Fig. 12

5.3.4 Formulario

El usuario rellena el formulario con cierta información personal, con fines únicamente estadísticos. El formulario consta de un tabla con scrooll (la edad), dos botones de tipo radio (sexo y trastorno) y un campo de texto para rellenar el NIA. Una vez cumplimentado el formulario se comienza el experimento clickando en “empezar”. La fig. 13 muestra la pantalla que se presenta al usuario.

Formulario

Por favor rellene el siguiente formulario, recuerde que es anónimo, sus datos solo serán utilizados con un fin estadístico:

Rango de edad

Sexo

NIA(Solo en el caso de cursar física en la UPNA)

¿Padece algún trastorno en la visión de los colores?

20-25

☐

Masculino

☐

Femenino

☐ si

☒ no

empezar

Fig.13

5.3.5 Experimento

En la Fig.14 se muestra la página del experimento. El usuario debe elegir la imagen que mas le gusta de las dos según su combinación cromática clickando en la elegida. El color de fondo es un gris al 50 % ya que es considerado un color neutro, de esta forma no influye en la elección del usuario.

Esta página se ejecuta variando las imágenes mostradas hasta que el usuario ha ordenado su lista completa, como vimos anteriormente aproximadamente unas 15 veces.

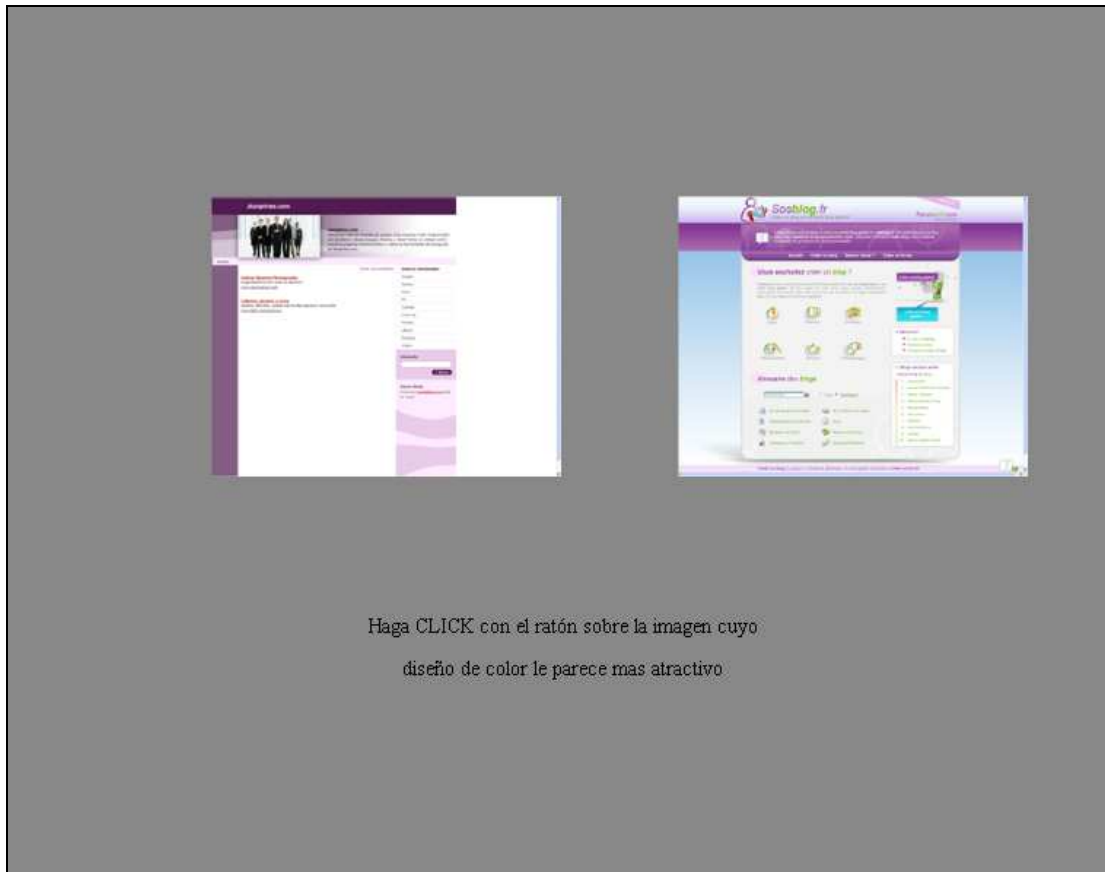


Fig.14

5.3.6 Feedback

En la retroalimentación o feedback se presenta al usuario las imágenes que ordenadas según sus elecciones y las imágenes ordenadas según la preferencia media de todos los usuarios. También se da la opción de repetir el experimento o volver al índice. En la Fig.15 se observa el feedback que se presenta a un usuario para su lista de imágenes.

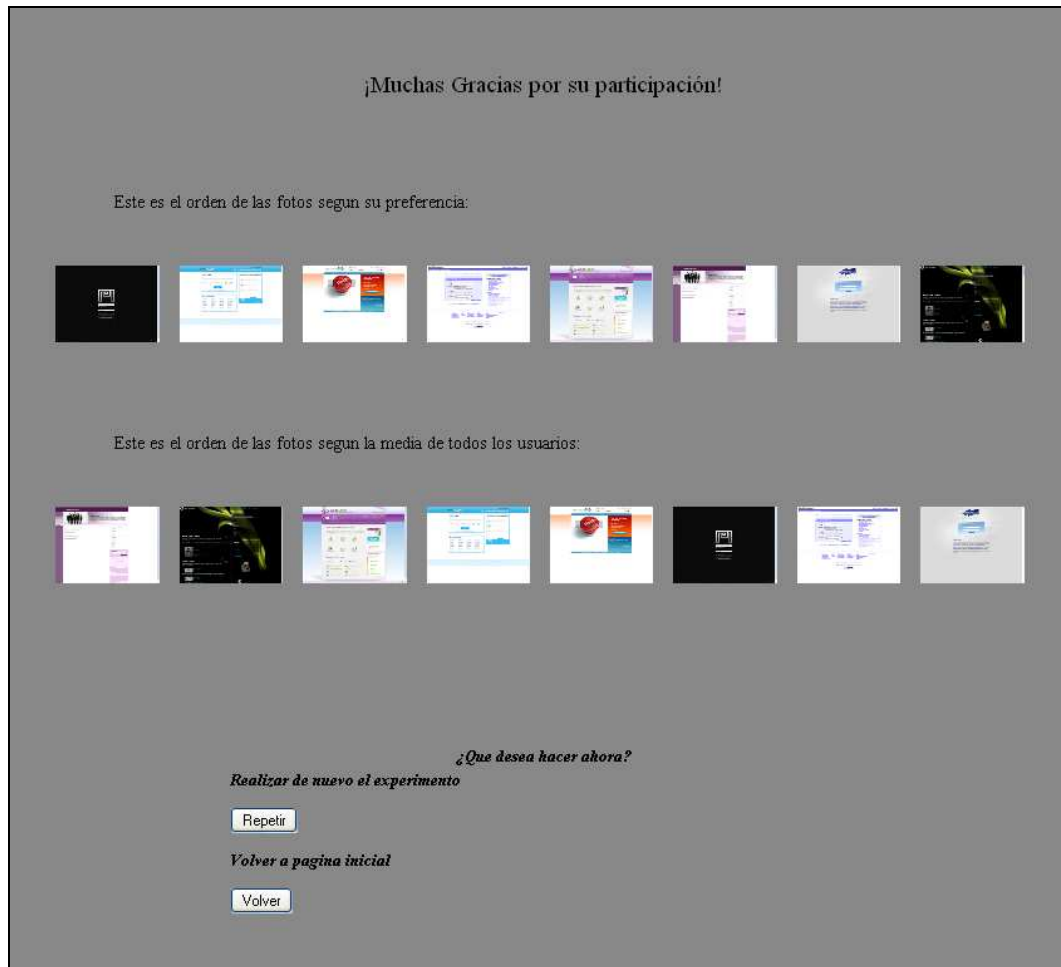


Fig.15

Como curiosidad podemos deducir que en la Fig.14 el usuario eligió la imagen de la derecha como preferida, y es por ello que en su lista queda por delante que la imagen de la izquierda, sin embargo la media de todos los usuarios prefiere la imagen de la izquierda sobre la de la derecha.

5.4 Almacenamiento

El almacenamiento de datos se realiza de diversas maneras, por un lado se guardan en archivos de textos los nombres de las imágenes con su asociación a cada número y la lista con las listas de imágenes desordenadas que cada usuario debe realizar; por otro lado se emplea una base de datos instalada en el servidor con el fin de guardar los datos estadísticos de cada usuario, la respuesta de dichos usuarios al test y otras variables que veremos a continuación.

5.4.1 Lista de fotos

Queremos que cada usuario tenga una lista de fotos aleatoria y diferente, de tal forma que cada 8 usuarios completen las 64 fotos, pero a cada usuario se le deben presentar una combinación de 8 imágenes diferentes. Para esto creamos una lista de la siguiente manera, desordenamos aleatoriamente una lista del 1 al 64, tomamos los primeros 8 valores para un usuario, los siguientes 8 para el siguiente y así hasta completar las 64 fotos, luego repetimos el proceso, volvemos a desordenar aleatoriamente la lista y así sucesivamente hasta obtener una lista lo suficientemente grande para abarcar una gran cantidad de usuarios.

Este proceso se realiza en el archivo “generalista.php”, que genera 4000 series de 8 números que cumplen las condiciones anteriores.

Estas son algunas líneas del archivo de texto “listas.txt” generado por “generalista.php” y que se usará para otorgar a cada usuario una lista diferente de fotografías:

```
17-42-2-43-57-28-38-7-  
47-39-1-20-59-61-45-9-  
55-51-35-29-34-32-63-52-  
23-31-30-10-15-25-14-4-  
56-27-22-37-33-19-8-53-  
11-62-21-60-13-50-12-64-  
24-16-41-54-49-36-48-58-  
6-5-46-26-3-44-18-40-
```

...

Supongamos que la primera línea que vemos (17-42-2-43-57-28-38-7-) es la línea 6 del archivo “listas.txt”, si el usuario al iniciar el experimento tiene un $m=6$ quiere decir que su lista de imágenes será la de la primera línea, al finalizar su experimento el valor “m” se incrementará en 1, de tal forma que el usuario siguiente tendrá un $m=7$ y su lista de imágenes será la 55-51-35-29-34-32-63-52-.

La lista que asocia cada foto con un número se guarda en el archivo “fotos.txt”, este archivo de texto está guardado dentro del servidor y cada línea de texto va asociada a una imagen, desde la línea 1 hasta la 64. Cuando se quiera representar una imagen cogeremos el número asociado a la imagen y el número de línea del archivo será la imagen asociada a ese número. Por ejemplo si en la aplicación estamos manejando la foto numero 5, vamos al archivo fotos.txt y vemos que nos estamos refiriendo a la foto “Small_guarantybank_com_I1.BMP”

Archivo “fotos.txt”:

```
Small_funbox_com_I1.BMP
Small_blog_co_in_I1.BMP
Small_pagesjaunes_ca_I1.BMP
Small_bes_pt_I1.BMP
Small_guarantybank_com_I1.BMP
Small_hotelscombined_com_I1.BMP
Small_gameforge_de_I1.BMP
...
```

La primera línea de este archivo está en blanco, ya que PHP considera la primera línea de un archivo como la línea 0.

5.4.2 Base de datos

La base de datos es una herramienta imprescindible para la realización de esta aplicación, en ella se guardan los datos estadísticos de los usuarios, los resultados obtenidos en los tests y variables auxiliares para la aplicación Web.

Para administrar la base de datos se utilizó “phpMyAdmin”, este permite administrar la base de datos de forma visual y gráfica desde el propio servidor interno. Para ellos se crea una cuenta con privilegios, que permite la creación y modificación de bases de datos y tablas.

La base de datos utilizada en este proyecto es “DBGonzalo”.

Conexión con base de datos

El usuario se conecta a la base de datos en una cuenta anónima y sin privilegios. La conexión se realiza en el archivo “conexion.php”, este utiliza principalmente dos funciones de php que permiten la comunicación con la base de datos, uno para conectarse a la base de datos:

```
$conexion = mysql_connect($dbhost, $dbusuario, $dbpassword) ;
```

El otro para seleccionar la base de datos “DBGonzalo”:

```
mysql_select_db($db, $conexion) ;
```

Todo ello va acompañado de sus respectivos mensajes de error en el caso de que fallaran.

Tablas de las bases de datos

En la base de datos “DBGonzalo” almacenaremos los datos en forma de tablas, son tablas de tipo MyISAM. Para cada tabla se define el número de campos, cada campo recibe un nombre. Para cada campo se especifica los tipos de datos, si el campo puede estar vacío (NULL o NOT NULL), el valor predeterminado si lo hubiese y la longitud máxima del campo.

Los datos estadísticos de cada usuario para el posterior análisis de resultado se guardan en la tabla “Infousuarios”, las respuestas al experimento de cada usuario se guardan en “SeleccionUsuario”, la lista con todas las fotos y sus respectivas puntuaciones se guardan en “Fotos” y el valor auxiliar “\$m” se guarda en la tabla “lista”.

Infousuarios

“Infousuarios” está compuesta por 9 campos que son: id, Sexo, NIA, Edad, m, trast, idioma, ip y fecha.

El campo “id” sirve para identificar cada usuario, es un numero entero que es diferente para cada nuevo registro de usuario, se define como un entero autoincrementable, es decir, cada vez que se hace un nuevo registro en la base de datos sin especificar el valor de “id”, este se incrementara en uno.

El campo “Sexo” puede tomar los valores “m”, “f” o ningún valor(masculino, femenino o NULL), el campo “NIA” contendrá el NIA en el caso de que los usuarios sean alumnos del tutor de este proyecto; el NIA esta definido como un cadena de números enteros de tamaño máximo el tamaño del NIA, así evitamos que los valores incorrectos de NIA introducidos queden registrados, también está definido como NULL.

El campo “Edad” registra la edad del usuario, esta puede tomar valores del 15 al 65 en saltos de 5 en 5. El campo “m” contiene la variable auxiliar que hace de puntero para la elección de la lista de 8 imágenes en el archivo “listas.txt”. “trast” puede valer “s” o “n” (si tiene trastorno en la visión de los colores o no tiene trastorno).

El idioma se guarda en el campo “idioma” tomando como valores “ESP”, “ENG” o “FRA”(Español, Ingles o Francés). La ip del usuario se guarda en el campo “ip” y la fecha en el campo “fecha”.

En la Fig.16. se muestra un registro cualquiera en la tabla “Infousuarios”.


	id	Sexo	NIA	Edad	m	trast	idioma	ip	fecha
	160	f		20	72	n	ESP	217.172.65.154	20-04-2010

Fig.16

Con la tabla “Infousuario” cubrimos los datos de cada usuario, pero no tenemos las elecciones que cada usuario ha hecho en la aplicación; para ello creo una tabla que llamo “SeleccionUsuario”.

SeleccionUsuario

En “SeleccionUsuario” encontramos el campo “id”, las fotos que le ha tocado al usuario, la puntuación que ha dado a cada foto, el tiempo mínimo, máximo y medio de respuesta y el tiempo total de realización de experimento (sin incluir la lectura de instrucciones ni rellenar el formulario).

El valor del campo “id” es el que relaciona la tabla “Infousuario” con esta tabla, es decir para cada usuario asignamos un numero que se guarda en “id”, cada usuario guarda su información en dos tablas, su información estadística en “Infousuario” y la información sobre sus respuestas en “SeleccionUsuario”, el conector entre ambas tablas es el “id” que será el mismo en las dos tablas.

En el campo “foto1” se guarda el numero que se asocia a la foto que mejor ha valorado el usuario, este va acompañado de “valor1” que toma un valor de 8. La segunda foto mejor valorada se guarda en “foto2” acompañada de su correspondiente valor en “valor2” y así sucesivamente hasta “foto8” y “valor8”. Todos estos valores son de tipo “float”, es decir pueden tomar decimales, el tiempo esta medido en segundos con dos décimas.

En “tiempotot” se guarda el tiempo que le cuesta al usuario realizar el experimento, en “tiempomin” el tiempo mínimo que le ha costado en la elección de una foto u otra, en “tiempomax” el máximo y en “tiempomed” el tiempo de elección promedio.

En la Fig.17 se muestra un registro de la tabla “SeleccionUsuario”, este registro es el mismo usuario que el mostrado en la Fig. de la tabla “Infousuario”, podemos ver como comparten el mismo “id”.

←T→	id	foto1	valor1	foto2	valor2	foto3		
<input type="checkbox"/>	160	56	8	1	7	13	/	...
valor3	foto4	valor4	foto5	valor5	foto6	valor6		
6	18	5	9	4	47	3	/	...
foto7	valor7	foto8	valor8	tiempotot	tiempomin	tiempomax	tiempomed	
41	2	44	1	158.37	2.67	25.98	9.9	

Fig.17

Fotos

La tabla “Fotos” es la encargada de guardar las veces que cada foto se ha mostrado al usuario, los puntos totales que lleva cada foto y la media de puntos.

Esta tabla se compone de cuatro campos: El campo “foto” donde está el número de foto, el campo “puntos” donde están los puntos totales que lleva cada foto, el campo “vistas” que

contiene el numero de veces que ha sido vista cada foto y el campo “media” que guarda la puntuación media de cada foto.

En este caso el funcionamiento de esta tabla difiere del de las dos anteriores, en esta tabla cada vez que el usuario realiza el experimento no se realiza un nuevo registro sino que se actualiza el registro ya existente.

Antes de comenzar a realizar el experimento a usuarios se crean 64 registros para cada campo, cada uno de estos registros equivale a una foto y se inician los valores de “puntos”, “vistas” y “media” en 0. Cuando un usuario realiza el experimento la aplicación toma el numero de foto y donde el campo “foto” vale ese numero se actualizan los campos , el campo “vistas” se incrementaría en 1, al campo “puntos” se le sumarían los nuevos puntos otorgados por el usuario para esa foto y el campo “media se actualizaría realizando la división entre “puntos” y “vistas”.

En la Fig.18 se muestran los registros de la foto 31 y la foto 32.





←T→			foto	vistas	puntos	media
<input type="checkbox"/>			31	8	43	5.38
<input type="checkbox"/>			32	6	43	7.17

Fig.18

Podemos observar en el ejemplo de la Fig.18 como la foto número 31 ha sido mostrada a 8 diferentes usuarios con una puntuación total de 43 puntos, esto nos da que de promedio la foto tiene una puntuación de 5.38, la foto 32 tiene los mismo puntos pero menos vistas por lo que su puntuación media es mayor. Si un nuevo usuario realizara el experimento y le tocará la foto 32 como parte de su lista de 8 imágenes y le diera la puntuación mínima (1 punto) este registro se actualizaría de la siguiente manera: “vistas” valdría 7, “puntos” valdría 44 y la media bajaría a 6.28.

lista

En la tabla “lista” se guarda la variable auxiliar “m”, al igual que “Fotos” el usuario no crea un nuevo registro en esta lista sino que modifica el valor del ya existente.

Esta tabla solo dispone de un campo, este campo se llama “m” y es un valor entero. Este numero va asociado a la lista de imágenes que hay que presentar al usuario.

Tenemos una gran lista con series de 8 números para presentar al usuario, cada vez que un nuevo usuario inicia el test el programa toma una nueva serie de 8 números, que a su vez van asociados a 8 imágenes. Podríamos decir que el valor “m” es el puntero de esa gran lista y cada vez que el usuario finaliza el test el puntero se incrementa en 1.

5.5 Funcionamiento de la aplicación

En este apartado se muestra el funcionamiento de la página, explicando brevemente las rutinas implementadas para el funcionamiento de la aplicación Web. El orden de explicación es similar al llevado en el aspecto visual de la página (véase 5.3).

5.5.1 Idiomas

Para que el experimento pueda servir en usuarios de múltiples nacionalidades la aplicación Web puede ser vista en castellano, inglés y francés.

Elección de Idioma

El idioma se puede elegir desde la página de portada del proyecto “index.php” o bien desde la página con las instrucciones “instrucciones.php”.

La página de portada está en castellano e inglés y da opción de elegir idioma para el siguiente paso.

En la página de la portada se utiliza un formulario HTML, con método GET y acción a la página de instrucciones, un selector HTML donde aparecen los diferentes idiomas cada uno asociado a su valor de idioma (\$ESP-Castellano, \$ENG-English,\$FRA-Français) y un botón “submit” que al clickarlo te dirige a la página de instrucciones pasando la variable idioma.

```
<form method="GET" action="instrucciones.php">
  <div align="center"><em>Eliga su idioma / Select your Language:
</em>
    <select name="idioma">
      <option value="ESP">Castellano</option>
      <option value="ENG">English</option>
      <option value="FRA">Français</option>
    </select>
  </div>

  <p align="center"> <input type="submit" name="Submit" value="Empezar /
Start"></p>
</form>
```

En la página de instrucciones lo primero que hace es guardar el valor enviado en la página anterior en la variable \$idioma. Si no estuviera definido lo pone por defecto en castellano.

```
if (isset($_GET['idioma'])) {
    $idioma=$_GET['idioma'];
}
```

```

}
else {
$id idioma="ESP";
}
include("idiomas.php");

```

En la misma página vuelve a haber otro selector de idiomas, en este caso la acción del formulario te dirige a la misma página con el nuevo valor de idioma elegido, volviendo al principio del código y guardando de nuevo la variable idioma con el nuevo valor. Como muestra el siguiente esquema.

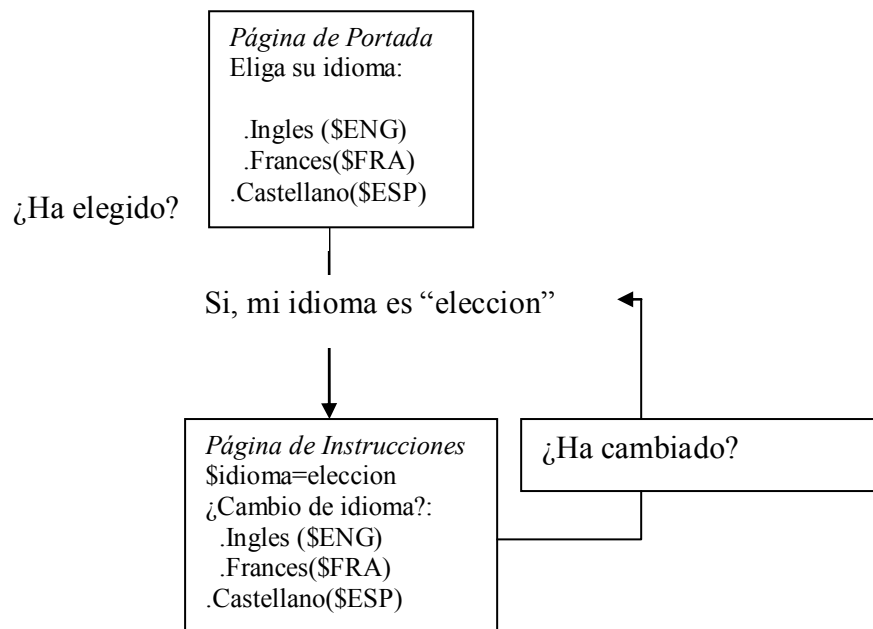


Fig.19

Funcionamiento

En PHP los vectores pueden ir asociados a índices de cualquier tipo ya sean números o caracteres.

Para poder cambiar de idioma fácil e instantáneamente hay un vector para cada idioma, cada elemento de estos vectores es texto de la página en el idioma asociado a ese vector, los índices de cada elemento del vector son un pequeño texto indicador del significado del texto, como muestra el siguiente ejemplo:

Castellano = “gat”=>”gato”

Ingles= “gat”=> “cat”

En este caso “Castellano” e “Ingles” serían los vectores que contienen el texto en castellano e ingles respectivamente, “gat” sería el índice que se asocia la palabra gato y a su traducción en inglés.

En “textos.php” se encuentran los vectores de cada idioma, todos los vectores tienen los mismos índices y a cada índice va asociado el texto correspondiente en el idioma correspondiente, como podemos apreciar en este trozo de código de “textos.php” (*véase apéndice código*):

```
$ESP= array( "selectidioma"=> "Selecione su
idioma" , "cambiar"=>"cambiar" ,
....

$ENG= array("selectidioma"=>"Select your language" ,
"cambiar"=>"change" ,
....
```

Aquí \$ESP es el array con el texto traducido a castellano y \$ENG es el array con los textos traducidos a ingles, el texto traducido a francés está en el array \$FRA.

Con todo esto consigo que para escribir el texto “selecione su idioma” en inglés pueda escribir el elemento “selectidioma” del vector \$ENG.

\$ENG[“selectidioma”]= select your language

En “idiomas.php” está la función “traduce(\$texto,\$idioma)”, esta función tiene como parámetros de entrada el idioma elegido y el índice del texto que se quiere escribir, “traduce” escribe el texto asociado a ese índice en el idioma elegido.

El funcionamiento de “traduce(\$texto,\$idioma)” es muy sencillo es un simple SWITCH con la variable \$idioma, si es ingles (case \$ENG) coge el vector \$ENG y el elemento asociado a \$texto

Codigo:

```
$idioma=$ENG;  
traduce("selectidioma",$idioma);
```

Resultado:

Select your language

Como la variable \$idioma se guarda al inicio del código tras ser elegido en la página de portada, como se explicaba anteriormente, ya se puede configurar todas las páginas sin escribir texto y guardando todo el texto como elementos de los vectores de cada idioma.

Todo el texto que contiene la aplicación Web está escrito a través de PHP, concretamente de la función “traduce” incluida en “idiomas.php”, este a su vez incluye “texto.php” donde está todo el texto en todos los idiomas guardado.

5.5.2 Portada e Instrucciones

La página inicial de la aplicación, que podemos ver en la Fig. 10, te introduce a los experimentos que ofrece COLORLAB como grupo de investigación del departamento de física.

En la página se encuentra el logotipo de la Universidad Pública de Navarra y el del grupo de óptica. Tras una breve introducción sobre el grupo COLORLAB y sus objetivos se ofrece al usuario realizar los experimentos desarrollados por el grupo, hay dos experimentos, el experimento 1: “Preferencias de diseño de color en páginas Web” es el que nos ocupa en este proyecto.

La pagina inicial recibe el nombre de “index.php” esto se debe a que al acceder a una URL la página que se abre es siempre la que recibe este nombre. Ofrece la posibilidad de cambio de idioma desde esa misma página.

Para entrar en el experimento se puede clicar en el texto del experimento o en el botón de empezar.

Al clicar en “experimento 1” te lleva al directorio “Pr/limpio/portada.php”, en el directorio “Pr/limpio” se encuentran todos los archivos “php” que componen la aplicación, así como las imágenes.

En “portada.php” se encuentra el título de proyecto, un selector de idioma y el botón de comenzar, se puede ver la página en la Fig. 11. Este botón te lleva a la página “instrucciones.php”, desde “portada.php” mandaremos por URL el idioma y lo guardaremos en la variable \$idioma, esto determinará el idioma en el que se presenten las instrucciones, desde las instrucciones también podremos volver a cambiar de idioma.

Estas páginas contienen poco código en php, exceptuando lo relacionado con idiomas. En “instrucciones.php” utilizamos un “isset” para comprobar si se ha modificado el idioma:

```
if (isset($_GET['idioma'])) {
    $idioma=$_GET['idioma'];
}

else {
    $idioma="ESP";
}
```

Si el idioma no ha sido elegido se elige el castellano como el idioma por defecto, después en el formulario selector de idioma, marcamos como idioma elegido el que esta guardado en la variable \$idioma, de la siguiente manera:

```
<select name="idioma" >
<option value="ESP" <?php if($idioma==ESP){echo selected;
}?>>Castellano</option>
<option value="ENG"<?php if($idioma==ENG){echo selected;
}?>>English</option>
```

Es decir si el idioma seleccionado sería ingles, \$idioma valdría ENG y por tanto la opción seleccionada (selected) sería la de “English”.

Todas la página de instrucciones utiliza la función “traduce”(vease 5.6.2 funcionamiento) para escribir el texto.

La ultima palabras de las instrucciones (“aquí” en castellano), es un hiperenlace al formulario.

5.5.3 Formulario

El formulario tiene una doble misión por un lado tomar los datos estadísticos del usuario de la aplicación y por otro lado iniciar las variables para el funcionamiento de la aplicación, en la fig.13 se muestra el aspecto visual del formulario, en la fig.20 se muestra un pequeño esquema del funcionamiento de “formulario.php”.

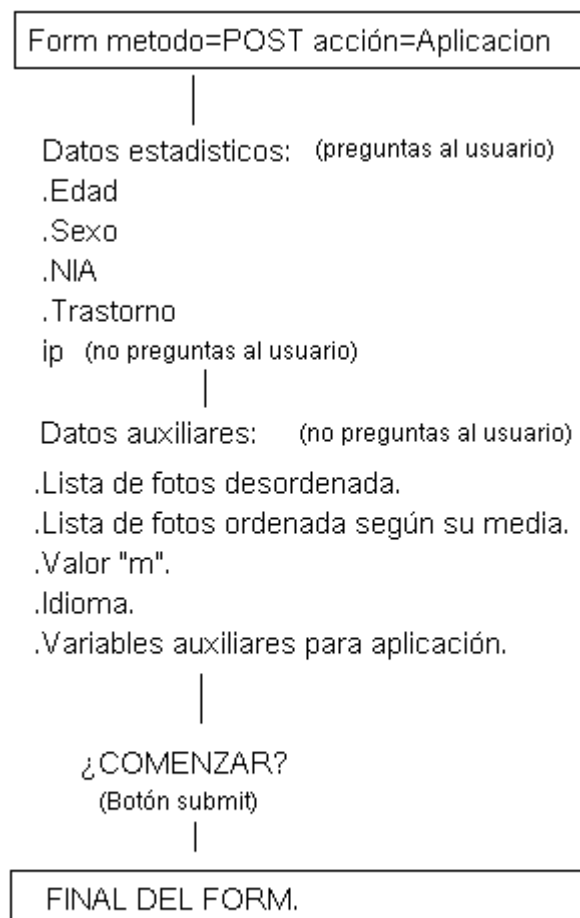


Fig.20

Para ello creamos un formulario HTML de tipo POST con acción a iterativo, como el siguiente código.

```
<form method="post" action="iterativo.php">
```

Como vemos en la Fig. al clicar el botón de comenzar todos los datos obtenidos ya sea preguntando al usuario o no(desde la base de datos) serán mandado a la acción del formulario, en este caso a “iterativo.php”

Datos estadísticos

En el formulario se pide al usuario una serie de datos estadísticos, estos son la edad, el sexo, el NIA y si padece algún tipo de trastorno en la visión de los colores.

La edad se toma mediante un scroll con intervalos de edad entre “<15” y “>65” con intervalos de 5 en 5, es decir de “15-20”, “20-25”, etc. Para ello utilizo un formulario de tipo “select” y la función java “MM_jumpMenu” que te permite hacer un menú desplegable.

Para cada valor del intervalo se le ajusta a su límite inferior, en caso de que el usuario no defina su edad esta se guardará con un “0”, si el usuario tiene entre 15 y 20 la edad se guardará como 15, a la hora de realizar las estadísticas consideraremos esto. Como se puede apreciar en este trozo de código:

```
<select name="Edad" onChange="MM_jumpMenu(this,0)">
  <option value=0>&nbsp;</option>
  <option value="10">&lt;15</option>
  <option value=15>15-20</option>
  <option value=20>20-25</option>
  <option value=25>25-30</option>
```

El sexo solo puedo tomar dos valores, masculino o femenino, también podría darse el caso de que el usuario no elija valor, en cuyo caso se guardara con un valor nulo en la base de datos. Para ello utilizamos un formulario de tipo radio con los dos posibles valores, masculino tomará el valor “m” y femenino el valor “f”.

```
<label>
  <input type="radio" name="Sexo" value="m">
  <?php traduce("masc", $idioma); ?>
</label>
```

Como podemos ver en el código los valores de cada etiqueta también cambian para cada idioma.

El NIA es un dato opcional solo lo tienen que introducir aquellos usuario que son alumnos de la UPNA. Para el NIA utilizamos un campo de texto con un tamaño máximo de 12 caracteres, en la base de datos solo se guardarán valores de NIA con un tamaño de 5 caracteres (el tamaño del NIA).

Por último se pregunta al usuario si padece algún trastorno en la visión de los colores, la entrada será de tipo radio tomando “s” como valor si el usuario tiene padece algún trastorno y “n” si el usuario no tiene. Como la mayoría de la gente no padece trastornos en la visión de los colores (entorno al 8% de la población padece algún trastorno en la visión de los colores) contestará que no, marcamos el no como respuesta seleccionada por defecto.

```
<label>
  <input type="radio" name="Trast" value="no" checked >
  <?php traduce("no", $idioma); ?>
</label>
```

El botón de comenzar es un input de tipo “Submit”, esto hace que al clickarlo te mande a la página que determina el “action” del formulario.

```
<input type="submit" name="Submit" value="<?php
traduce("empezar",$idioma); ?>">
```

Pero antes del botón de enviar datos y comenzar experimento la página coge otros datos que el usuario no ve, datos estadísticos y también datos que sirven para el funcionamiento de la aplicación.

El dato estadístico que se toma sin preguntar al usuario es la IP, esta se toma mediante una función de “php”.

```
$ipOrigen = $_SERVER['REMOTE_ADDR'];
```

Así guardamos en \$ipOrigen la ip del usuario, después la mandaremos a través de un “input” de tipo “hidden”, esta herramienta la utilizaremos mucho para mandar variables sin mostrarle al usuario.

```
<input name="ip" type="hidden" value="<?php echo $ipOrigen; ?>" />
```

Datos auxiliares

Una vez rellenado el formulario se pasa directamente a la aplicación, esta como después veremos se va llamando a sí misma de forma iterativa, por tanto conviene iniciar las variables que la aplicación requiera antes de entrar en ella.

Por esto en el formulario se mandan otro tipo de datos a parte de los estadísticos; se manda el valor “m”, la lista de fotos desordenada, la lista de fotos ordenada según su media, el idioma y una serie de variables auxiliares.

El valor “m” lo tomamos de la base de datos, para ello se accede a la base de datos incluyendo el archivo “conexión.php”, se realiza la consulta en la tabla “lista” y se guarda el valor en la variable \$m.

```
include("conexion.php");
$consulta= " SELECT `m`
FROM `lista`
LIMIT 1 " ;

$resultado= mysql_query($consulta , $conexion) or
die(mysql_error());

$sql_datos=mysql_fetch_array($resultado);
(integer)$m= $sql_datos["m"];
mysql_close($conexion);
```

En el formulario se incluye el archivo “cogelista.php” que es el encargado de coger la lista de fotos que se presentará al usuario en función de su valor “m.”

Para coger la lista no se requiere la base de datos, basta con acceder al archivo “listas.txt” con la función de PHP “file” y acceder a la línea que marca “m”. Después con la función “explode” separamos cada número delimitado por los guiones y lo guardamos en la variable \$lista; esta variable será la que contenga la lista de fotos a mostrar al usuario, guardaremos los valores de \$lista en el array \$desordenado, el elemento 0 de este array corresponderá a la primera imagen de la lista y el elemento 7 a la última.

```
//abro el archivo
$vlineas = file("listas.txt");
//selecciono la línea según m
$lista=$vlineas[$m];

//separo cada elemento
$lista= explode("-", $lista);
//indexo cada elemento en el array desorden
for($i=0;$i<8;$i++){
$desorden[$i]=intval($lista[$i]);
}
```

Para seleccionar la media de cada foto accedo de nuevo a la base de datos, incluyo el archivo “conexion.php”, realizo la consulta en la tabla “Fotos”, selección la foto y la media para cada foto, de esta forma se me guarda la media indexada para el valor de cada foto. La consulta en la base de datos es la siguiente:

```
$i=0;
$consulta= "SELECT `foto`,`media`
FROM `Fotos`
WHERE `foto` = ".$desorden[$i++]. "
OR `foto` = ".$desorden[$i++]. "
OR `foto` = ".$desorden[$i++]. "
OR `foto` = ".$desorden[$i++]. "
OR `foto` = ".$desorden[$i++]. "
OR `foto` = ".$desorden[$i++]. "
OR `foto` = ".$desorden[$i++]. "
OR `foto` = ".$desorden[$i++]. " ;
```

Utilizo de nuevo el “mysql_fetch_array” para pasar los datos obtenidos en un array, de esta forma consigo el array “\$med” que contiene los valores de las medias indexados con los valores de las fotos.

```
while ($sql_datos=mysql_fetch_array($resultado)) {
$med[$sql_datos['foto']]=$sql_datos['media'];
}
```

Después utilizamos la función “arsort” esta función ordena de mayor a menor un array manteniendo la correlación de índices, de esta manera ordenamos \$med según sus medias.

```
arsort($med);
```

Lo único que necesitamos es el array con la lista de fotos ordenadas para mostrar al usuario por tanto creamos un nuevo array que contenga solo los índices del array \$med.

```
foreach ($med as $key => $val) {  
    $media[$i++]=$key;  
}
```

Por ejemplo pongamos que nuestra lista de fotos es la siguiente:

\$desordenado=(8,23,9,7,4,21,13,60)

El vector \$med sería de la siguiente forma:

**\$med = [8=>2.5, 23 =>3.4, 9=>6.2, 7=>5.3, 4=>5.02, 21=>4.81,
13=>7.51, 60=>1.81]**

Despues de arsort:

**\$med =[13=>7.51, 9=>6.2 , 7=>5.3, 4=>5.02, 21=>4.81, 23 =>3.4,
8=>2.5, 60=>1.81]**

El array \$media quedaría finalmente:

\$media=[13,9,7,4,21,23,8,60]

El resto de las variables que se mandan en el formulario son auxiliares para el funcionamiento del mismo, pasamos por ejemplo el array ordenado vacío, contadores que se utilizan en el experimento, iniciados a 0 y otros índices que en el próximo punto veremos para que sirven.

5.5.4 Experimento

Toda la carga del experimento se encuentra en el archivo “iterativo.php”, aquí se muestra al usuario las imágenes, se calculan las imágenes siguientes a mostrar, se ordenan y una vez que se acaba se muestra el feedback al usuario y se almacenan todos sus datos en la base de datos.

En la Fig.21 podemos ver un esquema del funcionamiento general de “iterativo.php”.

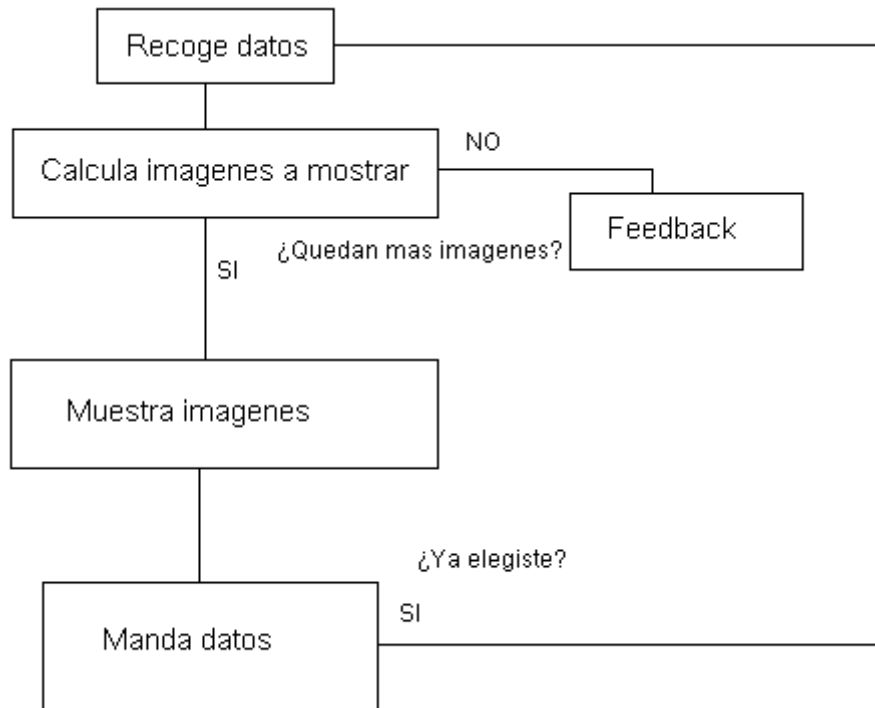


Fig.21

El experimento funciona de manera iterativa, se recogen unos datos al principio, estos datos son modificados y al final se vuelve a la misma página mandando los datos actualizados. La salida de este bucle se da cuando las 8 imágenes han sido ordenadas.

En el momento que se ha calculado las imágenes a mostrar la página que se muestra es la de la Fig. 14.

Recoge datos y manda datos

La recepción de datos se lleva a cabo en “recogedatos.php” y el envío de datos en “mandadatos.php”. Para mantener el círculo los datos que se mandan y se recogen deben ser los mismo; por ejemplo no puedes recoger un dato que no ha sido mandado, esto llevaría a errores.

Hay datos que se pasan de un lado a otro sin modificarse como la lista de fotos ordenada según su media y todos los datos estadísticos del usuario; los demás datos sufren modificaciones en función de la elección del usuario y de las necesidades de la aplicación.

La primera vez que se ejecuta “iterativo.php” los datos son recibidos del formulario, como ya explicamos anteriormente. Los arrays no pueden ser ni mandados ni recogidos enteros como arrays, hay que mandar elemento a elemento y luego reordenarlo en un array.

En “recogedatos.php” se recogen todos los elementos de la lista de fotos a mostrar al usuario, también se recogen los elementos de la lista ordenada, esta lista la manda inicializada a 0 el formulario y se va actualizando conforme el usuario va eligiendo una foto u otra.

La lista con las fotos ordenada según sus medias también es recogida en “recogedatos.php” y mandada al final del código en “mandadatos.php”.

Las variables \$i y \$j son los índices de la lista con fotos asociados a las fotos que deben mostrarse y la variable \$valor valdrá el numero de la foto \$i si el usuario elige la foto \$i y el numero de la foto de \$j si el usuario elige el la foto \$j, estos valores son recogidos actualizados y mandados de nuevo. Los valores de los tiempos (tiempomin, tiempomax y tiempotot) también recorren este circulo de envío de datos.

También circulan otra serie de variables auxiliares que mas tarde explicaré.

Mostrar Imágenes

Las imágenes están guardadas en el directorio del servidor “imágenes/” para mostrar una imagen solo hace falta poner un input de tipo imagen y escribir su directorio y el nombre de la imagen a mostrar.

```
<input type="image" src="imágenes/nombre_imagen">
```

Como antes hemos determinado los valores de \$i y \$j son los índices números de imágenes a mostrar. Es decir:

Foto[\$i]= Numero de imagen 1 a mostrar

Foto[\$j]= Numero de imagen 2 a mostrar

Para mostrar las imágenes teniendo el número asociado a ellas usamos un recurso parecido al usado para obtener la lista de fotos. Abrimos el archivo “listas.txt” que contiene la lista con el nombre de las fotos y lo guardamos en \$vlineas.

```
$vlineas = file("fotos.txt");
```

La primera imagen a mostrar corresponde a la línea \$i del archivo “listas.txt” y la segunda imagen a la línea \$j del mismo archivo. Tomamos de la variable \$vlineas las lineas de las imágenes a mostrar y las guardamos en las variables \$imagen1 y \$imagen2.

```
$imagen1=$vlineas[(integer)$desorden[(integer)$i]];
$imagen2=$vlineas[(integer)$desorden[(integer)$j]];
```

Una vez tenemos los nombres de las imágenes que queremos mostrar se añaden cada una a un formulario diferente en estos se mandan todos los datos, con una diferencia en el primer formulario el valor de \$valor toma el valor de la foto asociada a \$i y en el segundo formulario el de \$j.

El formulario para la primera imagen queda así:

```
<form action="iterativo.php" method="post">
<input type="hidden" name="pulsado" value="<?php echo
$desorden[(integer)$i]; ?>">
<input type="image" src="imagenes/<?php echo $imagen1; ?>" border='0'
alt="foto 1"/><?php include("mandados.php");?></form>
```

El formulario de la segunda imagen es similar al de la primera cambiando la foto mostrada y el valor del formulario “pulsado”.

```
<form action="iterativo.php" method="post">
<input type="hidden" name="pulsado" value="<?php echo
$desorden[(integer)$j]; ?>">
<input type="image" src="imagenes/<?php echo $imagen2; ?>"
border='0' alt="foto 2"/>
.....
<?php include("mandados.php");?>
```

Después en “recogedatos” se recogen todos los datos, en \$valor se guardará el número de la foto clickada:

```
$valor= $_POST['pulsado'];
```

Además de las imágenes se muestra un texto que indica al usuario que debe elegir una de las dos imágenes (vease Fig.14), este texto como en el resto de la aplicación utiliza la función “traduce()” para variar en función del idioma.

Ordena y calcula Imágenes

El motor de la aplicación se encuentra en el cálculo de las imágenes a mostrar y el ordenamiento según la elección del usuario. Como está dicho anteriormente este ordenamiento se basa en el algoritmo merge-sort para una lista de 8 imágenes desordenadas.

La función encargada de ordenar las listas y calcular el siguiente par de imágenes a mostrar es la función ordena_calcula(); Esta función tiene como parámetros de entrada los siguientes: La lista de fotos ordenada (\$ordenado), la lista de fotos desordenada(\$desorden), el

índice en la lista desordenada de las dos últimas imágenes que se han mostrado(i y j) y cuatro variables auxiliares para el funcionamiento de la función ($fase$, $cont1$, $cont2$ y $level$).

Esta función utiliza el paso de parámetros por referencia de tal forma que los cambios que se realicen dentro de la función sobre los parámetros de entrada modifican dichos parámetros.

La fase ($fase$) determina en que momento del ordenamiento se encuentra la función, si está ordenado por elementos simples la fase valdrá 1, si está ordenado por pares la fase valdrá 2 y si está ordenando por cuartetos la fase valdrá 3. Al iniciar “iterativo.php” la fase vale “0” y cuando acaba el ordenamiento la fase vale 4.

Al iniciar la página los valores de i y j están inicializados para mostrar las dos primeras imágenes de la lista($i=0, j=1$) y la fase está inicializada en 0, el resto de valores auxiliares están inicializados a 0. La función `ordena_calcula()` empieza con un switch en función de la fase, como se puede ver en este esquema la fase decide en que punto del ordenamiento se encuentra:

```
ordena_calcula(&$ordenado,&$desorden,&$i,&$j,&$valor,&$fase,&$cont1,&$cont2,&$level)
```

¿\$fase=0?

\$fase + 1

Sal de la funcion

¿\$fase=1?

ORDENAMIENTO SIMPLE

¿Fin del ordenamiento simple?

\$fase+1

\$i=0 y \$j=2

\$desorden=\$ordenado

Sal de la función

¿\$fase=2?

ORDENAMIENTO POR PARES

¿Fin del ordenamiento por pares?

\$fase+1

\$i=0 y \$j=4

\$desorden=\$ordenado

Sal de la funcion

¿\$fase=3?

ORDENAMIENTO POR CUARTETOS

¿Fin del ordenamiento por cuartetos?

\$fase+1

\$desorden=\$ordenado

Sal de la funcion

¿\$fase=4?

FEEDBACK

Sal de la función

Fig.22

Al iniciar “iterativo.php” se presentan al usuario dos imágenes sin que haya elegido anteriormente, estas dos imágenes son elegidas sin depender del usuario, son siempre las dos primeras imágenes del array con las lista de fotos desordenadas. Por tanto los valores de \$i y \$j son mandados desde el “formulario.php” con los valores 0 y 1 respectivamente, por eso en el primer acceso a “ordena_calcula()” no hay ninguna imagen que calcular ni ordenar, porque ya están calculadas previamente por lo tanto se incrementa la fase y se sale de la función.

```
switch ($fase) {
    case 0:
        $fase++;
    break;
}
```

Una vez la fase toma el valor de 1 comienza el ordenamiento simple.

Ordenamiento Simple

El ordenamiento simple es muy sencillo, se comparan las dos primeras imágenes, la elegida se ordena como la mayor respecto a la otra, después se comparan las dos siguientes y así hasta finalizar la lista.

```
if ($valor==$desorden[(integer)$i]){

    $ordenado[(integer)$i]=$desorden[(integer)$i];
    $ordenado[(integer)$j]=$desorden[(integer)$j];
    }
if
($valor==$desorden[(integer)$j]){

    $ordenado[(integer)$i]=$desorden[(integer)$j];
    $ordenado[(integer)$j]=$desorden[(integer)$i];
    }
    $i = $i+2;
    $j = $j+2;
```

Pongamos como ejemplo que se ha elegido la foto “\$desorden[\$j]” sobre la foto “\$desorden[\$i]”, y que esta es la foto número 24 sobre la foto 41, en este caso valor valdría 24. Se cumpliría el primer “if” del código y en el array \$ordenado quedaría de la siguiente manera respecto a \$desordenado:

```
$desorden=[ ..., 41,24,...]
      ↑  ↑
     $i $j

$ordenado=[ ..., 24,41,...]
      ↑  ↑
     $i $j
```

Una vez ordenados un par de elementos los índices \$i y \$j se incrementan en 2 para ordenar las dos siguientes imágenes.

Cuando se completa el ordenamiento simple se aumenta la fase en 1 y se sobrescribe el array \$desorden como el \$ordenado, esto implica que ya tenemos un array listo para ordenar por pares.

Ordenamiento por pares

Cuando la fase es 2 se entra en el ordenamiento por pares, esto se hace llamando a la función fase2, que tiene como parámetros de entrada los mismo que “ordena_calcula()” y también pasa los valores por referencia.

Para esta función se utilizan una serie de variables auxiliares, la variable \$level indica si se está ordenando los dos primeros pares o los dos últimos pares, al principio esta inicializada a 0 y cuando se termina de ordenar los dos primeros pares se incrementa en 1 para ordenar los dos siguientes.

Al iniciar la fase 2 los índices \$i y \$j están inicializados en 0 y 2, al elegir uno de los dos el elegido es colocado en la primera posición, el índice de la foto elegida se incrementará en 1, en la siguiente elección la foto elegida se colocará después de la foto anterior y se incrementará también su índice, a si obtenemos la relación de ordenamiento para el level=0 y para el level=1, si elegimos la foto del índice \$i el código por que seguiría sería el siguiente:

```
if ($valor==$desorden[(integer)$i]){
    if ($level==0){
        $a=($i+$j-2);
    }
    if ($level==1){
        $a=($i+$j-6);
    }
    $ordenado[(integer)$a]=$desorden[(integer)$i];
    $i=$i+1;
    $cont= $cont+1;
}
```

Cuando se elige la foto del índice \$j es lo mismo solo que se guarda en \$ordenado la foto del índice \$j y este se incrementa en 1. En el caso del índice \$j se incrementa el contador \$cont2.

Cuando se ha elegido dos veces los elementos de una de las listas ordenadas la otra lista se finaliza el ordenamiento del par de listas, puesto que la otra sublista ya esta ordenada.

Cuando se ha elegido una vez la imagen del índice \$i se incrementa el valor de \$cont y cuando se ha elegido la imagen del índice \$j se incrementa \$cont2, cuando uno de estos llega a 2 significa que ya podemos ordenar la lista, colocando los valores restantes de la otra sublista a continuación de estos.

Cuando se finaliza el último ordenamiento la fase se incrementa y se inicializan los valores de los índices \$i y \$j para el ordenamiento por cuartetos.

Ordenamiento por cuartetos

Cuando llegamos a este punto en \$desorden se encuentran dos sublistas de 4 elementos ordenados, ya solo queda ordenar estas dos sublistas, para ello se comparan los elementos de cada sublista, de tal forma que cuando el ultimo valor de una sublista sea elegido por el usuario el resto de la otra sublista se ordenaran automáticamente detrás de este último valor al estar ordenada previamente.

En este trozo de código se ve el proceso cuando se elige una de las imágenes y el ordenamiento final en el caso de que la primera sublista sea la que se ordene después del último elemento comparado:

```
if ($valor==$desorden[(integer)$i]){
    $a=$i+$j-4;
    $ordenado[(integer)$a]=$desorden[(integer)$i];
    $i=$i+1;
    if($i==4){
        while ($j<8){
            $ordenado[(integer)$j]=$desorden[(integer)$j];
            $j++;
        }
        $fase=$fase+1;
        $i=0;
        $j=0;
    }
}
```

Una vez realizado el último ordenamiento la fase se incrementa en 1, valiendo 4. Esto hace activar el feedback y salir del ciclo iterativo.

```
if ($fase==4){
    include ("feedback.php");
    exit;
}
```

5.5.5 Feedback

El feedback o retroalimentación tiene una doble función por un lado aquí se enseña al usuario las fotos que se le han mostrado ordenadas según su preferencia y según la preferencia media de todos los usuarios. Por otro lado es en el feedback donde se guardan todos los datos del usuario y se actualizan los datos de la base de datos.

Primero se calcula el tiempo medio de elección de fotos, dividiendo el tiempo total entre las iteraciones que el usuario ha llevado acabo. Después accedemos a la base de datos y

obtenemos de la tabla “Fotos” los valores de “vistas” y “puntos” para cada foto del array \$ordenado. Esto se realiza en el php “vistasypuntos.php” incluido en el feedback; Una vez obtenidos estos valores los actualizamos, incrementando en 1 las vistas, y sumando su respectiva puntuación a cada foto en “puntos”. Después se actualizan estos datos en “meter.php” incluido a continuación de “vistasypuntos.php”.

Con los arrays obtenidos de las “vistas” y “puntos” de cada foto creamos el array \$nuevamedia, este contiene la nueva media a introducir en la tabla “Fotos”, \$nuevamedia se calcula dividiendo los puntos de cada foto entre sus vistas:

```
for($i=0;$i<8;$i++){
    $nuevamedia[$ordenado[$i]]=round($puntos[$ordenado[$i]]/$vistas[
ordenado[$i]],2);
}
```

Una vez calculada la nueva media la actualizamos en la tabla “Fotos” también en el archivo “meter.php”.

Se realiza para esto una consulta para cada foto:

```
for($j=0;$j<8;$j++){
    $i=$ordenado[$j];
    $consulta = "UPDATE `DBGonzalo`.`Fotos` SET `vistas` =
    $vistas[$i] , `puntos` = $puntos[$i], `media` = $nuevamedia[$i]
    WHERE `Fotos`.`foto` = $i ";

    $resultado= mysql_query ($consulta, $conexion) or
    die(mysql_error());
}
```

Se incrementa el valor de \$m y se actualiza en la tabla “lista”, si llega a un máximo de 4000 (la lista es de mas grande que este número) el valor de \$m se reiniciaría a 1.

En “infousuarios.php” se introducen en la tabla “Infousuarios” los valores estadísticos del usuario, el valor de “id” se introduce como nulo, al ser un valor de tipo autoincrementable la base de datos incrementará en 1 este valor. Después consultamos el último valor “id” de la tabla “Infousuario” y lo utilizamos para introducir los datos de selección del usuario en “selecciónusuario.php”.

Una vez actualizada la base de datos se asocia para el array \$ordenado y para el array \$media a cada número de foto el nombre de la imagen a la que va asociada, a través de la “fotos.txt”.

```
$vlineas = file("fotos.txt");

for($i=0;$i<8;$i++){
    $foto[$i]=$vlineas[(integer)$ordenado[$i]];
}

for($i=0;$i<8;$i++){
    $fotoM[$i]=$vlineas[(integer)$media[$i]];
}
```

El array \$foto contiene los nombres fotos ordenadas de mayor a menor preferencia según el usuario y el array \$fotoM contiene los nombres de las fotos ordenadas según la media; Ya solo queda presentárselas al usuario de forma ordenada.

La presentación de las dos listas de imágenes se lleva a cabo en “nuevofeed.php”, después se da la opción de repetir, en cuyo caso se vuelve a la página del formulario, esto se hace desde “repetir.php” también incluido en el “feedback.php”.

6 RESULTADOS

Una vez puesta a prueba la aplicación y comprobado que funcionaba perfectamente se procedió a vaciar la base de datos para que los nuevos registros se correspondan con usuarios de la aplicación y evitar la confusión con las numerosas pruebas realizadas con anterioridad.

La muestra del primer usuario se tomó el 20 de Abril del 2010, para la realización de las estadísticas se tomaron todos los usuarios que realizaron el test desde la fecha de inicio hasta el 30 del mismo mes.

En estos días se consiguió una muestra suficiente como para comenzar con los análisis.

6.1 Datos sin procesar

En los 10 primeros días con el experimento abierto al público se consiguió que 100 usuarios completaran el test, con esta cifra se comenzó con los análisis estadísticos.

En la Fig.23 se muestra el histograma del número de comparaciones, comparado con su distribución normal. Aquí podemos comparar estos valores con los que tomaba la complejidad para el algoritmo mergesort, para 8 imágenes $\Theta=24$, esto nos daba el peor caso posible, comparando con el histograma se observa que los peores casos de esta muestra están entorno a 19 iteraciones. El promedio de iteraciones que obteníamos era 14.9, en el caso de la muestra de los 100 usuarios el promedio de iteraciones es 15,89. En el mejor caso posible el usuario ordena su lista de imágenes en 12 iteraciones.

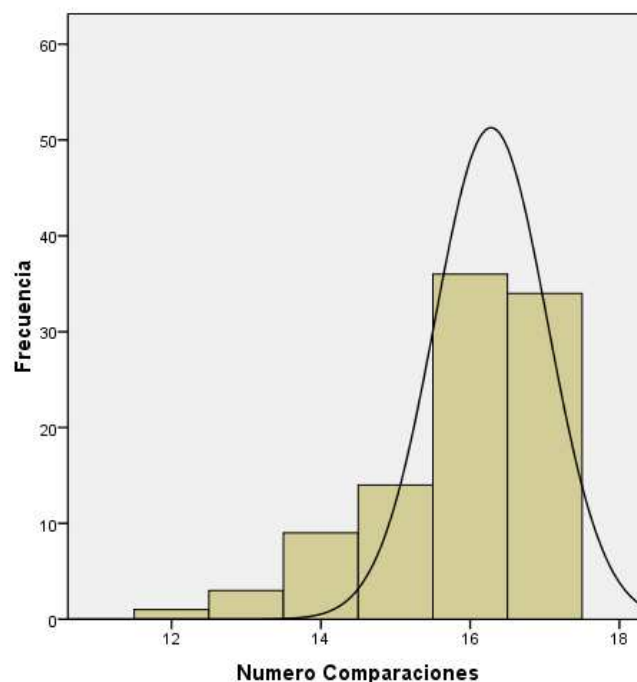


Fig.23

Las 100 muestras obtenidas tienen las siguientes características:

.50 son hombres, 48 mujeres y 2 no marcaron el sexo en el formulario.

.95 dicen no tener ningún tipo de trastorno en la visión de los colores y 5 dicen tener algún trastorno.

.La distribución de edades de la muestra sigue el siguiente esquema:

Edad	Frecuencia
0	2
<15	5
15-20	31
20-25	36
30-35	3
40-45	10
45-50	4
50-55	4
55-60	3
60-65	1
>65	1
TOTAL	100

En el caso de la edad, la edad igual a 0 supone que el usuario no ha introducido la edad antes de iniciar el experimento.

.99 usuarios han elegido el castellano para realizar el experimento, 1 usuario lo ha realizado en Inglés.

.En toda la muestra se repiten 18 IPs.

El siguiente paso es filtrar los datos, para obtener una muestra de usuarios válidos.

6.2 Filtrado

Con el proceso de filtrado se eliminan las muestras que carecen de contenido estadístico o que no son válidas para el análisis por diferentes motivos.

En este caso filtraremos a aquellos usuarios que no hayan introducido los datos de edad, o sexo en el formulario; a los usuarios con deficiencia en la visión de los colores y en función del tiempo de respuesta.

La repetición de IPs puede deberse a que varios usuarios han realizado el experimento desde el mismo terminal. Esto es algo normal debido a que la muestra es en muchos casos son familiares que realizan el experimento desde casa o estudiantes que se reúnen para realizar el experimento.

6.2.1 Tiempo de respuesta

Al finalizar el experimento se guarda en el registro el tiempo total de realización, el tiempo medio de elección entre dos imágenes, el tiempo mínimo de elección y el máximo. Estas variables tienen como función analizar la duración del experimento y de decisión y filtrar respuestas consideradas no válidas debido a un alejamiento de la función normal.

Para posibles anomalías en el tiempo de respuesta se analizan los histogramas de las distintas variables de tiempo y se comparan con una distribución normal de igual media y desviación típica a la distribución de la variable.

Si la diferencia entre el valor de la variable y la media es mayor que tres veces la desviación típica, es decir si “ $valor > \mu + 3\sigma$ ” ó “ $valor < \mu - 3\sigma$ ”, entonces el usuario de dicho valor será filtrado de la muestra.

Primero analizamos el histograma del tiempo total de ejecución de experimento, en la Fig.24 se puede ver este histograma comparado con la distribución normal. El promedio del tiempo total es 92,46 seg. y su desviación típica 51,02 seg. El tiempo total de 400 seg se sobresale bastante de la función normal. Analizando según el criterio de filtrado vemos que este valor se aleja de la media por 300 segundos, que es mas que tres veces la desviación típica; por tanto este dato se filtra.

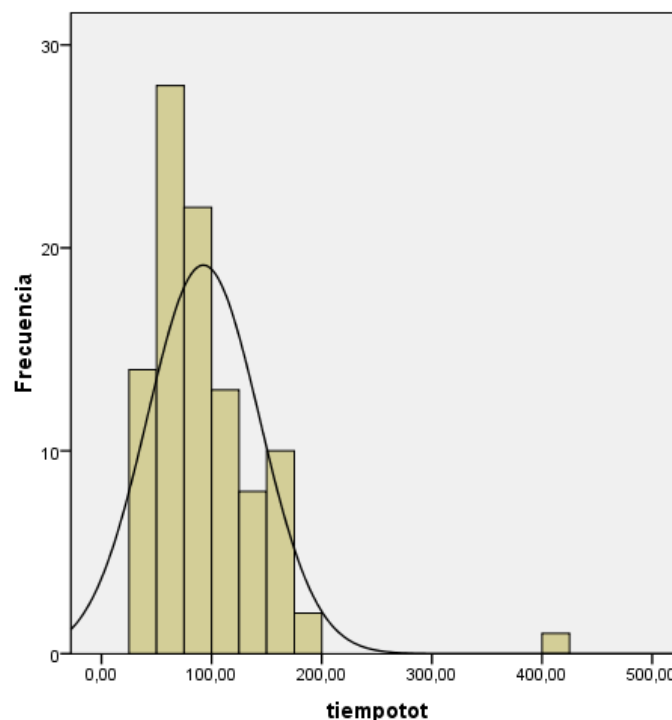


Fig.24

Una vez filtrado este dato se puede ver el histograma del tiempo total en la Fig. 25. Se puede ver que eliminando el dato que filtramos el histograma queda mucho más estirado y todos los valores forman parte de la distribución normal asociada a dicho valor. La media del tiempo total una vez filtrado es de 89,17 segundos y su desviación típica se reduce a 39.52 seg.

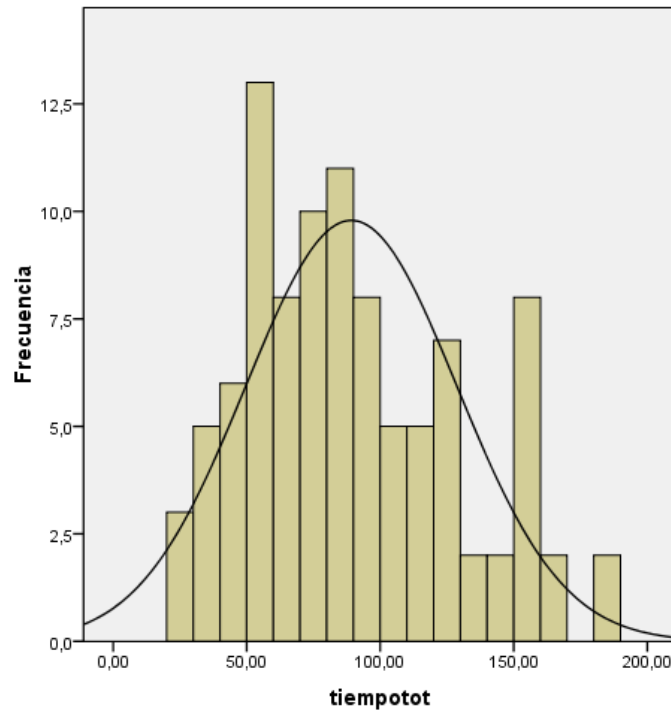


Fig.25

Una vez filtrado el dato de tiempo total realizamos el mismo proceso para los tiempos mínimos, máximos y medios.

El histograma para el tiempo mínimo esta representado en la Fig. 26. La media en el tiempo mínimo de elección es de 1,97 segundos, el mínimo está entorno a los 0.5 segundos y el máximo es de 5 segundos, la desviación típica es de 0.84 segundos. Comparando el histograma con la distribución normal asociada a este valor observamos que solo un dato se aleja de la normal. El valor que más se aleja es de unos 5 segundos, al comparar este valor con la media y su desviación típica obtenemos que no es lo suficientemente grande como para deber ser filtrado.

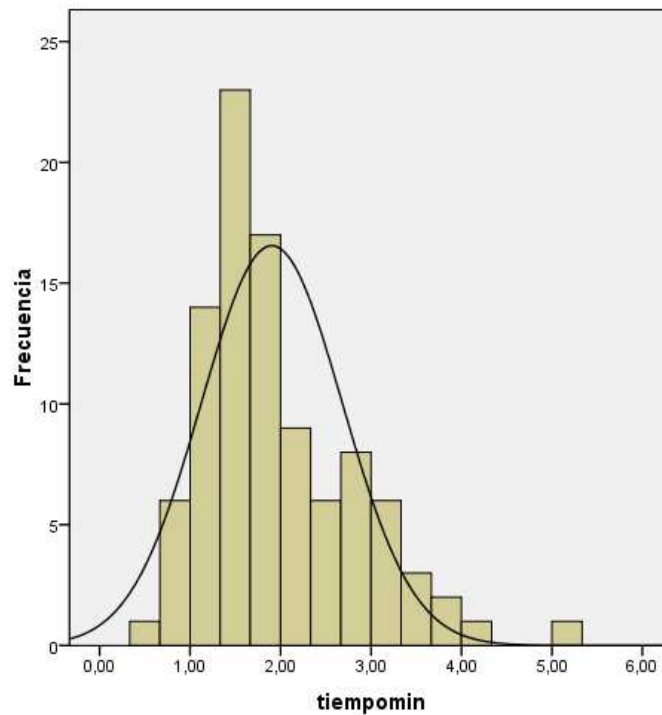


Fig.26

En la Fig. 27 se muestra el histograma del tiempo máximo, comparado también con su normal. La media del tiempo máximo es de 17.49 seg. y su desviación típica es de 14.77seg. El valor máximo está entorno a los 90 segundos.

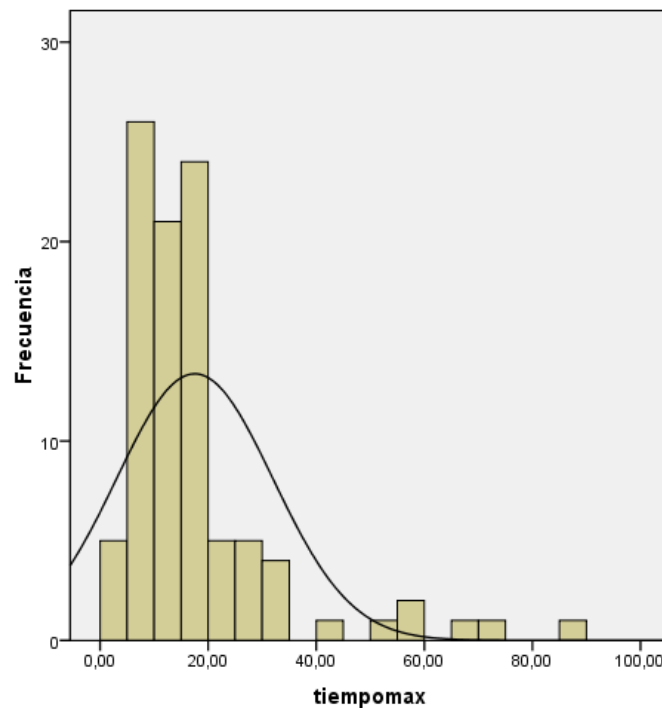


Fig. 27

En la Fig. 28 se muestra el histograma del tiempo medio de elección. La media del tiempo medio es de 5,62 seg. con una desviación típica de 2.48 seg. Los valores oscilan entre 1.8 seg. y los 12 segundos.

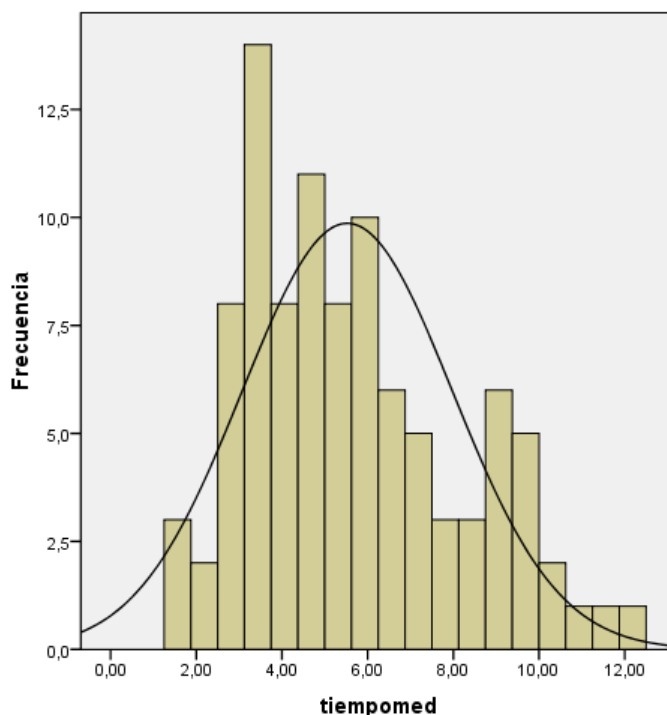


Fig.28

Al final por el filtrado en el tiempo hemos eliminado una muestra del total, considerándola no válida puesto que sobrepasa el tiempo total de ejecución del experimento.

6.2.2 Sin datos estadísticos

El experimento se puede realizar sin introducir datos estadísticos, simplemente eligiendo comenzar el mismo sin haberlos rellenado. Los usuarios que han realizado de dicha manera serán filtrados de la muestra de usuarios. De esta manera podemos realizar un estudio por grupos de edades y sexos.

Los datos a introducir de forma voluntaria son la edad y el sexo; El NIA es opcional y la selección de trastorno en la visión de los colores está seleccionada por defecto en “no”(puesto que la mayoría no padecen trastornos).

En el caso de edad filtramos todos aquellos usuarios que respondieron “0”, esta respuesta es la respuesta nula, asociada a que el usuario no eligió en el scroll su rango de edad; También filtraremos aquellos usuarios que no seleccionaron el sexo en el formulario,

6.2.3 Deficiencias en la visión del color.

Las deficiencias en la visión del color pueden ser congénitas o adquiridas, el primero de los casos es el más abundante y su porcentaje es considerable en el caso de los hombres, entorno a un 7-8 % de los hombres padecen de algún tipo de anomalía en la visión de los colores,

mientras que el porcentaje de las mujeres es mucho menor, entorno a un 0,5-0,6%. Este dato hace necesario preguntar a los usuarios si padecen algún tipo de trastorno.

Como se ha dicho antes 5 de los 100 usuarios afirman padecer algún tipo de trastorno, estos usuarios son filtrados de la muestra.

6.2.4 Estadísticos descriptivos después del filtrado

Una vez realizados estos tres filtros la muestra inicial de 100 usuarios queda reducida a 92, estos son los datos que quedan para iniciar con los análisis estadísticos:

.48 hombres y 44 mujeres.

.92 usuarios sin trastornos en el color.

.La distribución de edades de la muestra sigue el siguiente esquema:

Edad	Frecuencia
<15	3
15-20	29
20-25	34
30-35	3
40-45	10
45-50	4
50-55	4
55-60	3
60-65	1
>65	1
TOTAL	92

6.3 Resultados de la ordenación de las páginas Web.

La puntuación otorgada a las imágenes es del 1 al 8, siendo 8 la mejor puntuación y 1 la peor. Para calcular el orden de preferencia de las imágenes para los usuarios se divide el sumatorio de todas las puntuaciones para cada imagen entre las veces que ha sido mostrada cada imagen. Con esto conseguimos un valor medio del 1 al 8 que indica la preferencia media de los usuarios.

En esta tabla se pueden observar las puntuaciones que se han dado a cada foto, separado en hombres, mujeres y en total, las imágenes vienen ordenadas según el total:

Nº de imagen	Hombres	Mujeres	Hombres y Mujeres
32	7,16	6,86	7,02
12	6,20	6,00	6,10
31	6,50	5,50	6,02
40	5,33	6,67	5,97
29	6,60	5,17	5,91
20	5,11	6,67	5,85
25	5,00	6,75	5,84
35	6,66	4,87	5,81
14	5,68	5,86	5,76
56	7,11	4,00	5,62
1	4,71	6,50	5,57
22	5,33	5,80	5,56
53	6,43	4,60	5,55
59	5,43	5,67	5,54
7	5,20	5,87	5,52
28	5,50	5,54	5,52
6	5,00	5,75	5,36
9	5,00	5,75	5,36
54	3,83	7,00	5,35
36	5,80	4,83	5,34
34	5,75	4,71	5,25
51	5,67	4,67	5,19
61	5,71	4,60	5,18
27	5,00	5,33	5,160
57	5,17	5,00	5,088
50	5,25	4,75	5,01
64	6,43	3,20	4,88
2	3,67	6,00	4,78
11	4,80	4,75	4,78
48	5,00	4,25	4,64
52	4,67	4,56	4,61
37	4,75	4,40	4,58
4	3,83	5,14	4,46
45	5,00	3,86	4,45
60	4,50	4,25	4,38
63	4,86	3,80	4,35
8	3,14	5,67	4,35
16	2,71	6,00	4,27
26	6,00	2,29	4,22
41	4,00	4,25	4,12
47	3,67	4,40	4,08
13	4,00	3,71	3,86
58	3,12	4,67	3,86
17	3,33	4,33	3,81
44	2,80	4,86	3,78
46	3,50	3,80	3,64
62	4,00	3,25	3,64
49	2,20	5,20	3,63
5	4,17	3,00	3,61
43	3,33	3,80	3,56
24	3,86	3,00	3,45
19	4,40	2,33	3,41
3	3,75	3,00	3,39
21	3,86	2,67	3,29

33	3,33	3,00	3,17
23	2,67	3,50	3,06
42	3,25	2,86	3,06
39	4,14	1,75	3,00
10	3,00	2,67	2,84
55	3,12	2,33	2,74
38	2,57	2,80	2,68
30	2,25	2,33	2,29
15	1,40	3,12	2,22
18	2,75	1,00	1,91

En la Fig.29 podemos ver el orden de preferencia de los 92 usuarios del análisis. De forma que la imagen preferida es la 32, seguido de la 56 y la peor valorada es la imagen 15

En la Fig.30 están ordenadas según las preferencias de las mujeres de la muestra y en la Fig.31 según la de los hombres.

Para leer en la Fig. 29, 30 y 31, se coloca la hoja de forma horizontal siendo la imagen mejor valorada la de la primera línea a la izquierda y se lee de arriba abajo y de izquierda a derecha de mejor a peor respectivamente.

Orden de preferencia de las imágenes para el total de la muestra

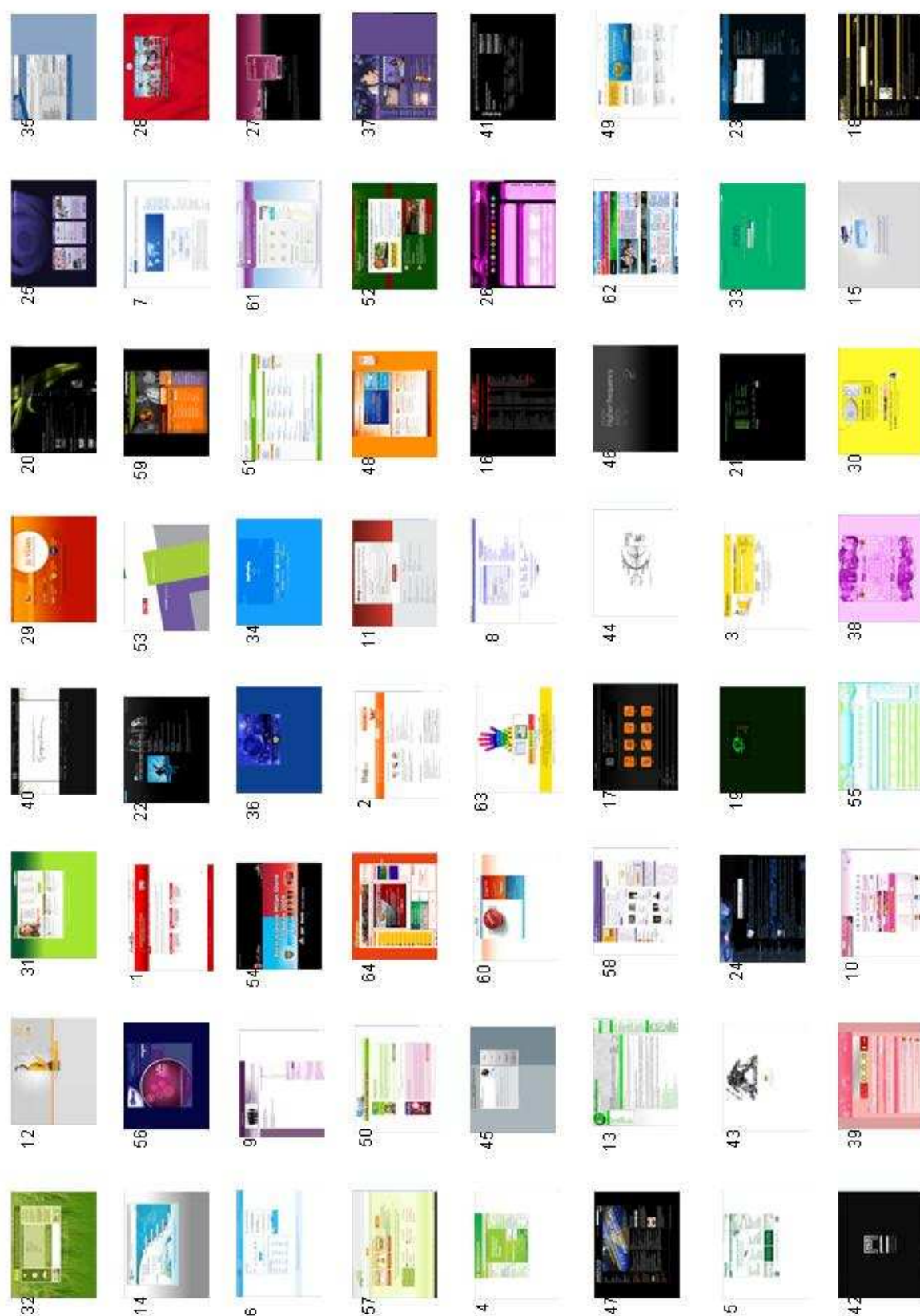


Fig.29

Orden de preferencia para las mujeres

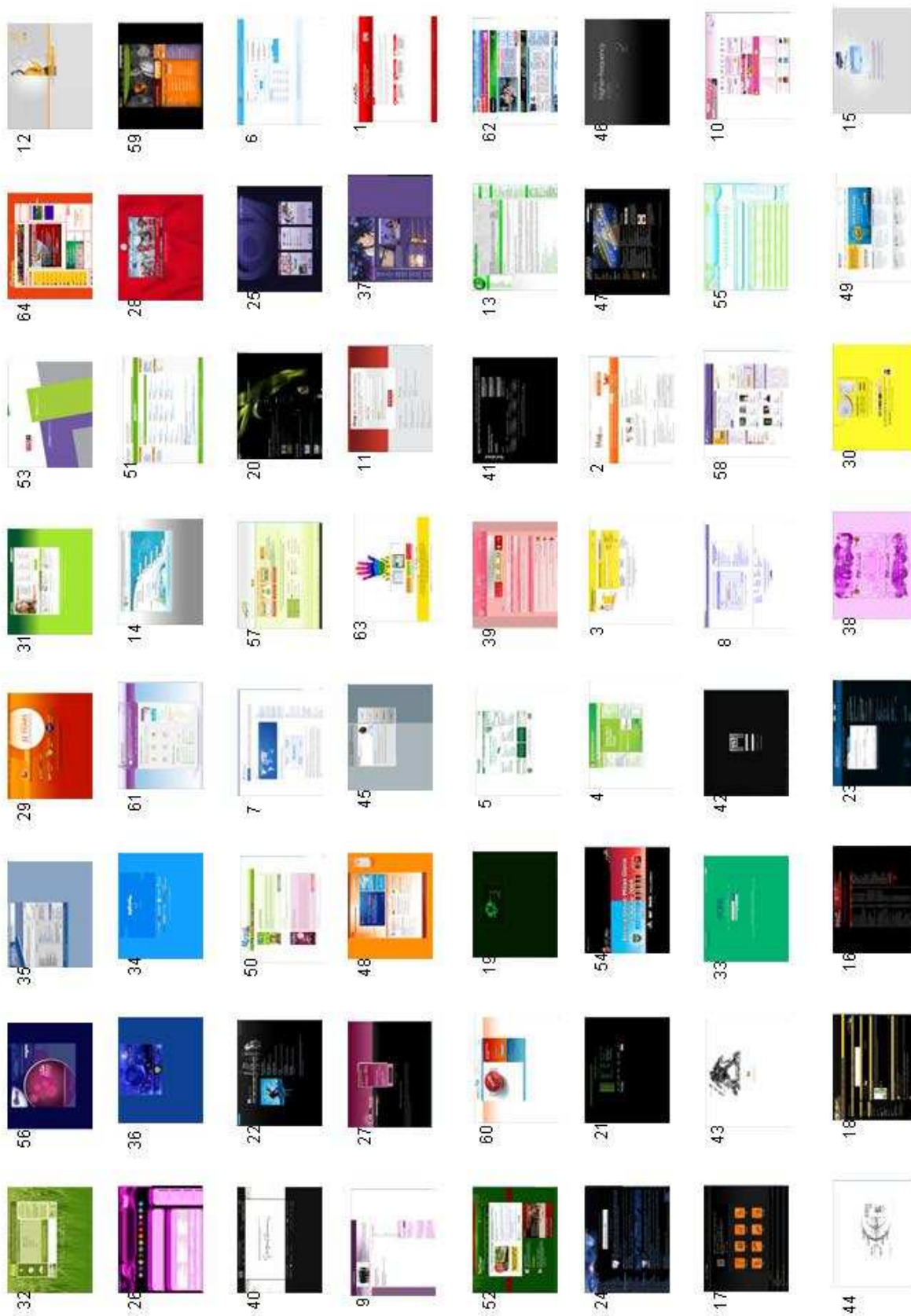


Fig..30

Orden de preferencia para los hombres



Fig.31

6.4 Análisis según criterios

Los análisis realizados han seguido los criterios de selección de imagen (véase 5.2.1) para ello se ha utilizado la estadística no paramétrica (véase 5.3).

Para todos los datos se analizará en la muestra total y dividiendo por sexos. Al no haber suficiente disparidad y cantidad de edades este parámetro no será utilizado para el análisis. En un futuro cuando la muestra aumente se podrá analizar con este estadístico.

6.4.1 Según claro-oscuro

El primer criterio de análisis va a ser la preferencia de los usuarios sobre las imágenes en función de luminosidad, para ello se utiliza el criterio de claro-oscuro que se mostraba en la Fig. 6, esto divide a las imágenes en tres clasificaciones: claras, medias y oscuras.

Para hacernos una idea más clara sobre la preferencia según luminancia de la que se obtiene visualmente de los resultados de la preferencia (véase Fig.29,30 y 31) se puede ver el histograma de la puntuación que la muestra ha dado a cada clasificación en la Fig.32,33 y 34, para las imágenes claras, medias y oscuras respectivamente. En el eje de abscisas está la puntuación entre el 1 y el 8.

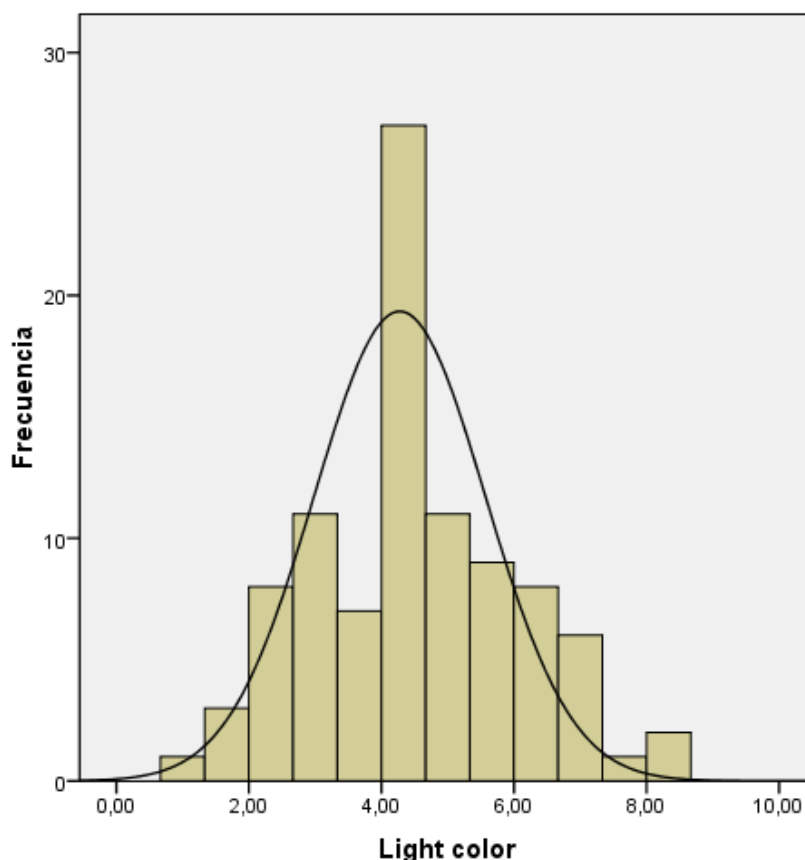


Fig.32

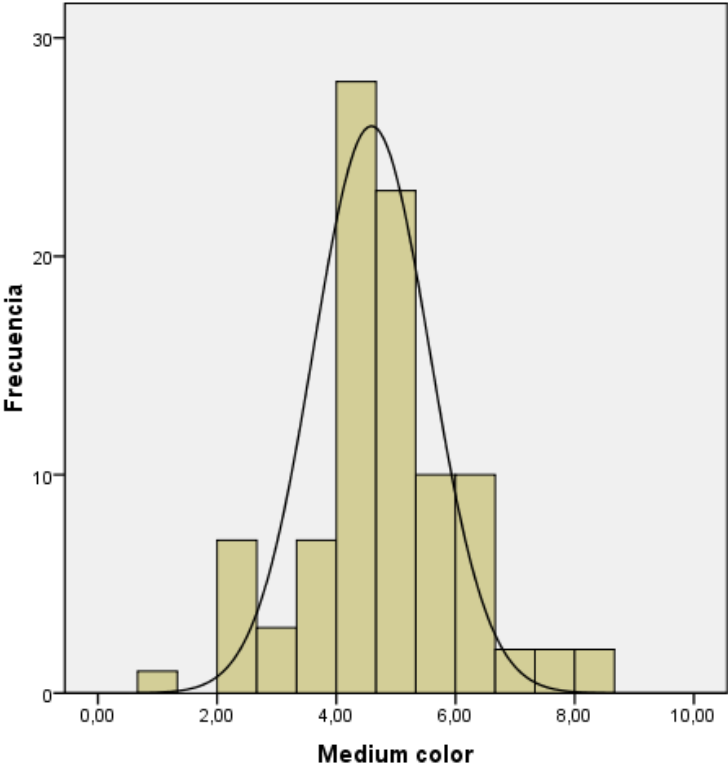


Fig.33

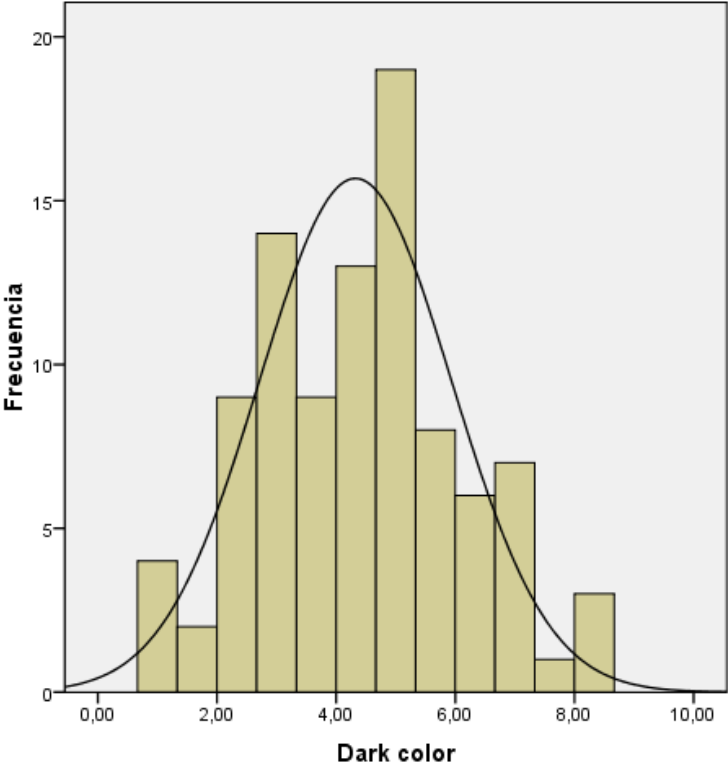


Fig.34

Es necesario que los usuarios sobre los que se va a analizar hayan tenido que elegir entre los tres tipos de imágenes, puesto que si solo tenían un tipo o dos de imágenes no podemos saber si las prefiere sobre el tipo de imagen que no se le ha mostrado. Partiendo de esta premisa se reduce la muestra de usuarios válidos a aquellos que hayan analizado en su lista los tres tipos de imágenes y obtenemos los siguientes resultados:

Tipos de Imágenes	Muestras	Media	Error típico	Desv. típica
Claras	87	4.30	0,17	1,58
Medias	87	4.65	0,15	1,38
Oscuras	87	4.419	0,18	1,73

La media indica la puntuación que se ha dado a cada tipo de imágenes dividido entre las veces que han sido votadas. Como podemos apreciar la diferencia de medias no es muy grande y considerando el error típico de la media se puede ver que todas las medias están próximas entre sí, lo que hace pensar que las diferencias de medias no serán significativas.

Al realizar la prueba de Friedman sobre la muestra el valor de significancia asintótica debe ser menor o igual a 0.05 para poder asegurar que las medias de cada tipo de imagen tienen una diferencia significativa.

Calculamos la significancia asintótica siguiendo la prueba de Friedman, aproximando los rangos promedios de las medias de las muestras a una Chi-cuadrado con 2 grados de libertad; Obteniendo una significancia de **0,438**. Esto nos indica que la **diferencia de medias no es significativa**, como ya anticipáramos. Al realizar la prueba de Kendall obtenemos el mismo nivel de significancia, extrayendo la misma conclusión.

Realizamos el mismo análisis dividiendo la muestra en hombres y mujeres.

Para las mujeres obtenemos los siguientes datos:

Tipos de Imágenes	Muestras	Media	Error típico	Desv. típica
Claras	44	3,99	0,25	1,65
Medias	44	4,93	0,20	1,34
Oscuras	44	4,47	0,27	1,77

En este caso ya se puede apreciar cierta diferencia entre la media de las imágenes claras y la media de imágenes medias. Aun sumando el error típico a cada media sigue existiendo diferencias entre estas, sobre todo entre las claras y medias.

Realizando la prueba de Friedman y la de Kendall obtenemos una significancia de **0,078**, **no es suficiente** para asegurar que la diferencia es **significativa** pero se acerca más al 0.05 necesario.

Para los hombres obtenemos los siguientes datos:

Tipos de Imágenes	Muestras	Media	Error típico	Desv. típica
Claras	43	4,62	0,22	1,47
Medias	43	4,3547	0,21	1,38
Oscuras	43	4,3605	0,26	1,71

Para los hombres se aprecia que **no** existe **diferencia** entre medias; al realizar la prueba de Friedman y la de Kendall se confirma esta sospecha dando una significancia asintótica de **0,892**, muy lejos de ser significativa.

Dado que en el caso de las mujeres si se ha observado que tiene cierta significancia la diferencia entre medias, se va a comprobar comparando entre una y otra categoría para concretar esta diferencia.

Para ello utilizamos la prueba de Wilcoxon, se elige las veces que se ha mostrado a las usuarias dos tipos de imágenes conjuntamente y se analizan sus respuestas en rangos. Obtenemos la siguiente muestra para realizar la prueba:

	Comparación	Muestra	Rango promedio
Claras-Medias	medias < claras	15	17.03
	medias > claras	28	24.66
	medias = claras	3	
Claras-Oscuras	claras < oscuras	19	20.42
	claras > oscuras	24	23.25
	claras = oscuras	2	
Oscuras-Medias	oscuras < medias	28	22.41
	oscuras > medias	17	23.97
	oscuras = medias	0	

Realizando la prueba de signos de Wilcoxon obtenemos que solo existe una diferencia significativa en el caso de “medias>claras” con un nivel de significancia **0.009**. Con lo que

podemos concluir que las mujeres prefieren las imágenes medias respecto a las claras con un probabilidad de error del 0.9%.

Para los hombres se ha realizado la misma prueba, con resultados **sin significación** estadística, por lo que no se puede concluir ninguna preferencia específica para ellos.

Como último análisis para este criterio se compara la diferencia en la preferencia según sexo para las tres clasificaciones. Es decir si el gusto por claras, oscuras y medias es diferente en función del sexo.

Para este último análisis se utiliza la prueba de Mann-Whitney, la muestra necesaria es las veces que se ha mostrado a cada sexo una de las clasificaciones, obtenemos la siguiente muestra:

Tipo de imágenes	Sexo	Muestra	Rango promedio
Claras	Hombre	43	50.91
	Mujer	47	40.55
	Total	90	
Medias	Hombre	44	40,56
	Mujer	47	51,10
	Total	91	
Oscuras	Hombre	44	45,10
	Mujer	46	45,88
	Total	90	

Obtenemos tras realizar la prueba que los gustos por páginas **claras y medias** difieren entre sexos con una significancia de **0.059** y de **0.056** respectivamente. Es decir se puede afirmar con una significancia casi suficiente que los hombres prefieren las imágenes claras respecto a las mujeres y con casi la misma significancia que las mujeres prefieren las imágenes medias respecto a los hombres. También se concluye que las preferencias sobre imágenes oscuras son similares en ambos sexos.

6.4.2 Según el fondo

El siguiente análisis que se realiza es en función del fondo de las imágenes. Podemos ver este criterio de clasificación de imágenes en la Fig.7 (véase 5.2.1.2). Las imágenes se dividen en 4 grupos según su fondo: blanco, negro, gris o cromático. Algunas imágenes quedan fuera de esta selección por no pertenecer a ninguno de estos grupos.

Como en el anterior criterio el primer paso es analizar si existen diferencias significativas entre los distintos grupos para toda la muestra, para ello se toman a aquellos

usuarios de la muestra que han comparado en su lista los 4 tipos de imágenes, siguiendo ese criterio obtenemos los siguientes datos:

Tipo de Imagen	Muestra	Media	Error típico	Desviación típica
Fondo Blanco	41	4	0,21	1,34
Fondo Negro	41	4,65	0,32	2,05
Fondo Gris	41	4,65	0,33	2,13
Fondo Color	41	4,97	0,27	1,73

Se observa en la tabla como existe una diferencia entre medias, concretamente la media en la preferencia de color sobre el blanco es considerable, aun con sus respectivos errores típicos.

Tras la realización de la prueba de Kendall y la de Friedman se obtiene que la diferencia entre todas las medias no es significativa.

Aunque las diferencias del total de la muestra no sean significativas si lo pueden ser al analizar en función del sexo, puesto que ciertas preferencias pueden quedar anuladas ante preferencias contrarias. Por tanto se realiza el mismo análisis para hombres y mujeres.

Para la mujer se obtienen los siguientes datos a analizar:

Tipo de Imagen	Muestra	Media	Error típico	Desviación típica
Fondo Blanco	19	3,82	0,29	1,27
Fondo Negro	19	4,25	0,44	1,91
Fondo Gris	19	4,42	0,50	2,19
Fondo Color	19	5,6	0,41	1,79

Aunque la muestra sea menor si que se aprecia a simple vista una diferencia entre los valores, la media de las imágenes de fondo de color es casi dos puntos mayor que las de fondo blanco. Al realizar la prueba de Friedman y la de Kendall se obtiene que si existe una diferencia significativa, con un nivel de significación de **0,046**.

Para los hombres se obtienen los siguientes datos a analizar:

Tipo de Imagen	Muestra	Media	Error típico	Desviación típica
Fondo Blanco	22	4,16	0,31	1,45
Fondo Negro	22	4,99	0,46	2,14

Fondo Gris	22	4,84	0,45	2,11
Fondo Color	22	4,42	0,32	1,5

En el caso de los hombres las diferencias entre las medias no son tan grandes a simple vista como las de las mujeres. Al aplicar sobre estas muestras la prueba de Friedman y la de Kendall se obtiene un valor de significancia insuficiente para afirmar que existen diferencias.

Estas primeras pruebas solo dicen si existen diferencias significativas entre las muestras pero no especifica cuales ni las cuantifica. Para analizar mas concretamente se realizan otras pruebas que comparan en función de un tipo de imagen u otro.

Como para los hombres ya da como resultado que no existen diferencias significativas no es necesario realizar este tipo de pruebas, de todas las formas se realizaron sin obtener ningún análisis concluyente.

Las imágenes de fondo gris son las que menos veces aparecen en la muestras, el error típico en las medias de las valoraciones de este tipo de imágenes es considerable y el gris es una combinación de blancos y negros, por estas razones eliminamos el fondo gris para realizar la comparación uno a uno; Con esto conseguimos aumentar la muestra de usuarios que han comparado en la misma lista los mismos fondos.

Eliminando el fondo gris de la comparación obtenemos la siguiente muestra de mujeres que han comparado dos de entre los otros tres tipos de fondos:

Tipos de Imágenes	Muestra	Media	Error típico	Desviación típica
Fondo color	47	5,33	0,25	1,73
Fondo negro	46	4,17	0,28	1,9
Fondo blanco	45	3,95	0,24	1,60

La comparación ha realizar va a ser entre las imágenes de fondo coloreado y las de fondo blanco y por otra parte las de fondo blanco y negro. Obteniendo los siguientes datos:

	Comparación	Muestras	Rango promedio
Fondo Blanco – Fondo Color	blanco < color	31	24,52
	blanco > color	13	17,69
	blanco = color	0	
Fondo Blanco - Fondo Negro	blanco < negro	22	22,41
	blanco > negro	20	20,50
	blanco = negro	1	

Realizamos la prueba de signos de Wilcoxon y obteniendo un valor de significancia del **0,002** para la hipótesis “blanco<color” lo que nos permite asegurar que las **mujeres prefieren** las imágenes de **fondo coloreado** sobre las de **fondo blanco** con dicho nivel de significancia. En el caso de la comparación blanco-negro no se obtiene ningún valor de significancia para ninguna de las hipótesis.

El último análisis en este criterio es ver la diferencia en la preferencia según sexo para las cuatro clasificaciones. Es decir si el gusto por los diferentes fondos es diferente en función del sexo.

Para ello en cada fondo se comparan las puntuaciones medias que los usuarios han dado a cada fondo cada vez que les ha tocado compararlo, en función del sexo. Obteniendo así los siguientes datos:

Tipo de imagen	Sexo	Muestra	Rango promedio
Fondo blanco	Hombre	43	49,03
	Mujer	45	40,17
	Total	88	
Fondo negro	Hombre	43	46,93
	Mujer	46	43,20
	Total	89	
Fondo gris	Hombre	26	25,48
	Mujer	23	24,46
	Total	49	
Fondo color	Hombre	42	37,19
	Mujer	47	51,98
	Total	89	

Al realizar la prueba de Mann-Whitney y la de Kruskal-Wallis sobre estos datos se obtiene un nivel de significancia de **0,007** para el fondo de color. Esto nos indica que existe una diferencia significativa entre la preferencia de los hombres y la de las mujeres sobre las imágenes de fondo de color y podemos concluir que a las mujeres les gustan mas las imágenes de fondo coloreado que a los hombres.

6.4.3 Según el tono

El último criterio de análisis es según el tono predominante de la imagen, la clasificación se puede ver en la Fig.8 y divide las imágenes en 8 clasificaciones (véase 5.2.1.3).

En la siguiente tabla podemos ver las distintas clasificaciones, la frecuencia de usuarios ha los que se le a presentado, la media de puntos y la desviación típica para cada color:

Tipo de Imagen	Muestra	Media	Desviación típica
Acromatica	54	4,10	2,12
Roja	37	4,97	2,09
Naranja	42	5,07	2
Amarilla	29	2,77	1,79
Verde	69	4,56	2,04
Azul	70	4,59	1,93
Violeta	31	4,79	2,11
Magenta magenta-roja	56	3,95	2,35

En esta tabla se puede ver como las imágenes mejor valoradas son las naranjas y las rojas y la peor valorada la amarilla. Al existir 8 clasificaciones diferentes es muy difícil encontrar usuarios que hayan comparado las 8 clasificaciones

El primer análisis va a ser comparar las preferencias según sexos, para ello se comparan las medias de cada imagen en función del sexo con la prueba de Wilcoxon y la de Kruskal-Wallis.

Los datos a utilizar son las medias para cada tipo de imagen según su tono y según el sexo, estos datos son los siguientes.

Tipo de Imagen	Sexo	Muestra	Rango promedio
Acromática	Hombre	28	28,38
	Mujer	26	26,56
	Total	54	
Roja	Hombre	20	22,95
	Mujer	17	14,35
	Total	37	
Naranja	Hombre	22	23,00
	Mujer	20	19,85
	Total	42	
Amarilla	Hombre	13	14,81
	Mujer	16	15,16
	Total	29	
Verde	Hombre	32	32,42
	Mujer	37	37,23
	Total	69	
Azul	Hombre	34	35,90
	Mujer	36	35,13
	Total	70	
Violeta	Hombre	12	18,92
	Mujer	19	14,16
	Total	31	
Magenta magenta-roja	Hombre	26	25,37
	Mujer	30	31,22
	Total	56	

Al realizar las pruebas antes dichas se obtiene que los hombres prefieren las imágenes **rojas** respecto a las mujeres con una significancia del **0,015**

Para realizar un análisis mas exhaustivo sobre la preferencia según tono de los usuarios es necesario no discernir entre sexos, puesto que de esta manera se obtiene una muestra mayor y por tanto se pueden encontrar más niveles de significancia necesarios para contrastar una hipótesis.

Con los datos obtenidos se va a realizar un análisis comparando todas las imágenes entre si, para ello se utiliza la prueba de signos de Wilcoxon. Se toman todas las posibles combinaciones y se analizan mediante esta prueba; por ejemplo se toma el naranja y se compara

con todos los colores, la muestra de esta comparación serán aquellos usuarios que han comparado el naranja con cada uno de los colores a comparar, después se toma el rojo y se compara con todos los colores excepto el naranja que ya ha sido comparado.

Mediante este método conseguimos obtener la significancia de todos los colores entre sí, en la siguiente tabla se muestra la significación de la diferencia de medias de la comparación de todos los tonos entre sí para toda la muestra de usuarios válidos:

En aquellas comparaciones en las que la diferencia no es significativa se indica con el símbolo “_____” en la casilla correspondiente.

Significancias en la preferencia de un Color1 sobre otro Color2									
Media	Color1\Color2	Naranja	Rojo	Violeta	Azul	Verde	Acromático	Magenta	Amarillo
5,07	Naranja		_____	_____	_____	_____	0,038	_____	0,018
4,97	Rojo			_____	_____	_____	_____	0,027	0,035
4,79	Violeta				_____	_____	_____	_____	_____
4,59	Azul					_____	0,008	_____	0,058*
4,55	Verde						_____	_____	0,028
4,10	Acromático							_____	0,014
3,96	Magenta								_____
2,77	Amarillo								
*No es significativo pero se aproxima.									

En esta tabla podemos asegurar con sus respectivas significancias que los usuarios prefieren las imágenes naranjas sobre las acromáticas y las amarillas, las rojas sobre las magentas o magentas rojas y amarillas, las azules sobre las acromáticas, las verdes sobre las amarillas y las acromáticas sobre las amarillas.

6.4.4 Conclusiones generales de los resultados

Después de realizar las estadísticas a todos los criterios de análisis a toda la muestra y dividiendo la misma en función del sexo hemos obtenido una serie de conclusiones.

En cuanto a la luminosidad de las imágenes y sus fondos hemos concluido que las mujeres prefieren las imágenes de luminosidad media respecto a los hombres y respecto a las imágenes claras; que los hombres prefieren las imágenes claras respecto a las mujeres; y que a las mujeres les gustan mas las imágenes de fondo coloreado respecto a los hombres y respecto a las de fondo blanco.

En cuanto a los tonos de las imágenes, las naranjas y las rojas son las preferidas por los usuarios, mientras que las amarillas, magentas y acromáticas son las peor valoradas. El resto de las conclusiones sobre los tonos de las imágenes se pueden ver de forma gráfica en la anterior tabla.

Existen otras conclusiones cuya significancia ha quedado cerca del valor límite para ser conclusiones estadísticas, estas son:

- Los usuarios prefieren las azules sobre las amarillas.
- A los hombres les gustan mas las imágenes claras que a las mujeres.
- Las mujeres prefieren las imágenes de luminosidad media respecto a los hombres.

En la Fig.25,26 y 27 están las imágenes ordenadas según la preferencia de los usuarios, de las mujeres y de los hombres respectivamente; Estas imágenes aportan visualmente la preferencia de los usuarios. Aunque no sea de manera estadística, se pueden obtener conclusiones viendo las imágenes mejor valoradas o las peor valoradas.

CONCLUSIONES

Para este proyecto se ha realizado una aplicación en entorno Web que permite el estudio de las preferencias del usuario respecto del diseño de color de las páginas y sitios Web

El experimento se realiza de forma automatizada y estandarizada, no requiere de un seguimiento por parte del investigador, este puede acceder a los registros en cualquier momento y analizar las respuestas dadas por los nuevos usuarios. Esto permite aumentar la muestra de usuarios que realizan la aplicación sin ningún esfuerzo por parte del investigador.

Así mismo, el aumento de la muestra estadística permitirá realizar análisis en función de otros datos como los rangos de edad.

Con los análisis estadísticos realizados sobre esta primera muestra de participantes de la aplicación del proyecto, se ha conseguido obtener las primeras conclusiones sobre la preferencia cromática de los usuarios en páginas Web. Se han establecido conclusiones sobre la preferencia de los tonos predominantes en las páginas Web; por ejemplo que los usuarios prefieren los tonos naranjas sobre los amarillos o los acromáticos, o que los usuarios prefieren los tonos verdes sobre los amarillos.

Se han obtenido también otras conclusiones dividiendo la muestra en subgrupos, como que las mujeres prefieren las imágenes de iluminación media frente a las claras. Se han comparado también dichos subgrupos para obtener diferencias significativas entre ellos y se han sacado conclusiones como que los hombres prefieren las imágenes rojas respecto a las mujeres.

Todas estas conclusiones obtenidas del análisis estadístico pueden servir a la hora de plantear el diseño de un sitio Web, puesto que dichas conclusiones tienen validez estadística y están realizadas sobre una muestra de imágenes pertenecientes a páginas Web reales con diferentes combinaciones cromáticas.

Finalmente hay que tener en cuenta que las conclusiones estadísticas obtenidas con los análisis de los resultados se deducen de un número limitado de participantes (aproximadamente 100).

En el momento de finalizar esta memoria el número de participantes de la aplicación es de 228 y es de esperar que en un corto espacio de tiempo se llegue a los 400 usuarios.

Si asumimos que las indeterminaciones estadísticas evolucionan con un factor \sqrt{n} , siendo “n” el número de usuarios; es de esperar que las incertidumbres en los diferentes análisis que se realizan se reduzcan en un factor 2, lo que nos permitirá un análisis con mayores conclusiones significativas.

ANEXOS

1 Fuentes

1.1 Bibliografía

Tim Converse y Joyce Park. “PHP 4 BIBLE”.

Robert W. Sebesta. “Programming the World Wide Web”.

Michael H. Birnbaum. “Psychological Experiments on the INTERNET”.

Edward J. Giorgianni y Thomas E. Madden. “Digital color management”.

1.2 Webgrafía

<http://es.wikipedia.org>

<http://ocwus.us.es/metodos-de-investigacion-y-diagnostico-en-educacion/analisis-de-datos-en-la-investigacion-educativa>

<http://www.chuidiang.com>

<http://www.desarrolloweb.com>

<http://www.hilite.me/>

<http://www.maestrosdelweb.com>

2 Código

2.1 *Index.php*

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN">

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<title>Colorlab</title>
<style type="text/css">
<!--
body
.Estilo6 {font-size: xx-large; color: #333333; }
a:link {
    color: #333333;
    text-decoration: none;
}
a:visited {
    text-decoration: none;
    color: #333333;
}
a:hover {
    text-decoration: none;
    color: #FF0000;
}
a:active {
    text-decoration: none;
    color: #CCCCCC;
}
body {
    margin-left: 0px;
    margin-top: 0px;
    margin-right: 0px;
}
-->
</style></head>

<body>
<table width="100%" border="0" cellpadding="0" cellspacing="0">
<tr>
<td background="design/nar.JPG"></td>
<td width="618"></td>
<td background="design/nar.JPG"></td>
</tr>
<tr>
<td><div align="center"></div></td>
```

```

        <td>&nbsp;</td>
        <td><div align="center"></div></td>
    </tr>
</table>
<p align="center">&middot; <a href="index2.php"><em>Castellano</em></a>
&middot; <a href="index.php">Euskara</a> &middot; <a
href="index3.php">English</a> &middot;
</p>
<p>&nbsp;</p>
<p>&nbsp;</p>
<blockquote>
    <table width="100%" border="0">
        <tr>
            <td><p>&nbsp;<b>Bienvenidos a <strong><font
size="4">COLORLAB</font></strong>, la página del grupo de investigación en
óptica del Departamento de Física de la Universidad Pública de Navarra. En
esta página te ofrecemos el poder participar en experiencias dedicadas al
estudio de la percepción del color. <br>
            Con tu participación contribuirás al estudio de algunos aspectos relativos
a la percepción del color, de forma sencilla y cómoda. Realizar los
experimentos que se proponen a continuación no te llevará más de 15
minutos.<br>
            En cada experimento te daremos unas breves instrucciones y al final te
proporcionaremos un pequeño resumen comparado de tus respuestas.</p>
            <p><br>
            Gracias por participar.</p>
            <p>&nbsp;</p></td>
        </tr>
    </table>
    <p>&nbsp;</p>
</blockquote>
<blockquote>
    <p align="center"><em>&nbsp;<b>Elija un experimento:</b></em></p>
    <table width="100%" border="0">
        <tr>
            <td><table width="350" height="75" border="1"
bordercolor="#FF3300"align="center">
                <tr>
                    <td width="335"><p align="center"><strong>Experimento
1:</strong></p>
                    <p align="center"><a href="/Pr/limpio/portada.php">Preferencias
de diseño de color en páginas Web.</a></p>
                    <form action="/Pr/limpio/portada.php">
                        <p align="center">
                            <input type="submit" name="Submit" value="empezar">
                        </p>
                    </form></td>
                </tr>
            </table>
            <td>
                <table width="350" height="75" border="1"
bordercolor="#FF3300"align="center">
                    <tr>
                        <td><p align="center"><strong>Experimento 2:</strong></p>
                    </td>
                </table>
            </td>
        </tr>
    </table>

```



```

        <p align="center"> <a href="/I/exp/index.php">Proporciones de
color y equilibrio cromático.</a> </p>
        <form action="/I/exp/index.php">
        <p align="center">

                <input type="submit" name="Submit" value="empezar">
        </form></td>
        </tr>
        </table>
        </td>
        </tr>
        </table>
        <p>&nbsp;</p>
</blockquote>
</body>
</html>

```

2.2 Portada.php

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"
/>
<title>Estudio de la percepción de los colores en páginas
Web</title>
</head>

<body>
<table width="800" border="0" cellspacing="2" cellpadding="0">
  <tr>
    <td width="153"><font size="5"><strong><font size="6"
face="Georgia, Arial" color="red"></font></strong></font></td>
    <td width="641"></td>
  </tr>
</table>
<p>&nbsp;</p>
<p>&nbsp;</p>
<p>&nbsp;</p>
<p>&nbsp;</p>

<p align="center"><font size="6"><strong><font size="7"
face="Georgia, Arial" color="red">E</font></strong><font
color="red">studio de la percepción de los colores en páginas
Web.</font></font></p>

```

```

<p align="center"><font face="Georgia, Arial" color="red">-Study of
color vision in Web pages.-</font></p>
<p>&nbsp;</p>
<p>&nbsp;</p>
<p>&nbsp;</p>
<p>&nbsp;</p>
<p>&nbsp;</p>
  <form method="GET" action="instrucciones.php">
  <div align="center"><em>Eliga su idioma / Select your Language:
&nbsp;</em>
    <select name="idioma">
      <option value="ESP">Castellano</option>
      <option value="ENG">English</option>
      <option value="FRA">Français</option>
    </select>
  </div>

  <p align="center">  <input type="submit" name="Submit"
value="Empezar / Start"></p>
  </form>
</body>
</html>

```

2.3 Instrucciones

2.3.1 Instrucciones.php

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Instrucciones</title>
<script language="JavaScript" type="text/JavaScript">
<!--
function MM_jumpMenu(targ,selObj,restore){ //v3.0
  eval(targ+ ".location='"+selObj.options[selObj.selectedIndex].value+"'");
  if (restore) selObj.selectedIndex=0;
}
//-->
</script>
</head>

<body>
<?php
if (isset($_GET['idioma'])){
    $idioma=$_GET['idioma'];
}
else {

```

```

$ididioma="ESP";

}

include("idiomas.php");

?>

<form action="instrucciones.php" method="GET">

<table width="100%" border="0" cellpadding="0" cellspacing="1">
  <tr>
    <td width="90%">&nbsp;</td>
    <td width="10%"><font size=2><em>
      <?php traduce("selectidioma",$idioma); ?>
    </em></font></td>
    <td><select name="idioma" >
      <option value=""ESP" <?php if($idioma==ESP){echo selected;
    }?>>Castellano</option>
      <option value=""ENG" <?php if($idioma==ENG){echo selected;
    }?>>English</option>
      <option value=""FRA" <?php if($idioma==FRA){echo selected;
    }?>>Français</option>
    </select></td>
  <tr>
    <td width="90%">&nbsp;</td>
    <td width="10%">&nbsp;</td>
    <td><p>
      <input type="submit" name="Submit" value="" <?php
    traduce("cambiar",$idioma); ?>" />
    </p></td>
  </tr>
</table>
</form>

<!-- aqui viene el texto inicial -->
<table width=100% height="671" border="0" cellpadding="0" cellspacing="0">
  <tr>
    <td width=15%>&nbsp;</td>
    <td width=70%>
      <?php traduce("textot1",$idioma); ?>
    </td>
    <td width=15%>&nbsp;</td>
  </tr>
</table>

<!-- aqui viene la imagen de las dos fotos -->

<table width=100% border="0" cellpadding="0" cellspacing="0">
  <tr>
    <td width=15%>&nbsp;</td>
    <td width=70%><p align="center"><img src=<?php imagenidioma($idioma); ?>
width="377" height="125" /></p></td>
    <td width=15%>&nbsp;</td>
  </tr>
</table>

```

2.3.2 texto.php

upna
Universidad
Pública de Navarra
Nafarroako
Unibertsitate Publikoa

Todos los derechos reservados
Eskubide guztiak erresalbatu dira

<p align=\"justify\"> En este experimento nos centraremos en la influencia del color. En concreto nuestro objetivo es determinar cuáles son las preferencias de los usuarios en cuanto al uso del color y de las combinaciones de colores en páginas Web.</p>

<h2 align=\"center\"><u>Desarrollo del experimento.</u></h2>

<p align=\"justify\"> La realización del experimento es muy sencilla. En primer lugar le pediremos algunos datos estadísticos (sexo, edad, etc) y seguidamente comenzaremos con el experimento.</p>

<p align=\"justify\"> Para conocer sus preferencias, el programa le mostrará dos imágenes, una al lado de la otra, como se muestra en la figura. Se trata de imágenes correspondientes a páginas Web reales.</p>

"texto2"=>"Haga CLICK con el ratón sobre la imagen cuyo",

"texto3"=> "diseño de color le parece mas atractivo",

"texto4"=> "<p> </p>

<p align=\"justify\"> Fíjese en las imágenes, centre su atención únicamente en el color y su distribución. No se deje influenciar por otros aspectos que no sean únicay exclusivamente el color.</p>

<p align=\"justify\">Dado que las imágenes se han obtenido de páginas Web reales es posible que conozca alguna de ellas. No se deje influenciar por este hecho. No se fije tampoco en los contenidos de la página. Concentre su atención en el uso del color que se hace en cada una de ellas.</p>

<p align=\"justify\"> Observe cada pareja de imágenes unos segundos, no emplee demasiado tiempo. Una vez que haya decidido cuál es la imagen que más le gusta, pulse sobre ella con el ratón.</p>

<p align=\"justify\"> Según vaya contestando, el programa le mostrará nuevas parejas de imágenes. Observará que las distintas imágenes se van repitiendo, aunque nunca le mostraremos la misma pareja dos veces. Esto nos permitirá conocer perfectamente sus preferencias.</p>

<p align=\"justify\"> Si ha leído y comprendido estas instrucciones, puede pasar a realizar el experimento pulsando ",

"aqui"=> "aquí",

"formulario"=>"Formulario",

"form"=> "Por favor rellene el siguiente formulario, recuerde que es anónimo, sus datos solo serán utilizados con un fin estadístico: ",

"edad" => "Rango de edad",

"sexo"=> "Sexo",

"masc"=> "Masculino",

"fem"=> "Femenino",

"nia"=> "NIA(Solo en el caso de cursar física en la UPNA)",

"trast"=> "¿Padece algún trastorno en la visión de los colores?",

"si"=> "si",

"no"=>"no",

"empezar"=>"empezar",

"image1"=>"Haga CLICK con el ratón sobre la imagen cuyo",

"image2"=>"diseño de color le parece mas atractivo",

"agradecer"=>"¡Muchas Gracias por su participación!",

"feed1"=>"Este es el orden de las fotos segun su preferencia:",

"feed2"=>"Este es el orden de las fotos segun la media de todos los usuarios:",

"rep1"=>"¿Que desea hacer ahora?",

```
$ENG= array(
```



```
"feed1"=>"Ceci est l'ordre des photos en fonction de votre
préférence:",
"feed2"=>"Ceci est l'ordre des photos selon la moyenne de tous les
utilisateurs:",
"rep1"=>"Que voulez-vous faire maintenant?",
"rep2"=>"Refaire le test",
"rep3"=>"Répéter",
"rep4"=>"Retour au département de physique",
"rep5"=>"Retour");

?>
```

2.3.3 idiomas.php

```
<?php

function imagenidioma($idioma){
    switch($idioma){
        case "ESP":
            echo "image_espanol.jpg";
            break;
        case "ENG":
            echo "image_ingles.jpg";
            break;
        case "FRA":
            echo "image_ingles.jpg";
            break;
    }
}

function traduce($texto,$idioma){
    include("texto.php");
    switch($idioma) {
        case "ESP":
            echo "$ESP[$texto]";
            break;
        case "ENG":
            echo "$ENG[$texto]";
            break;
        case "FRA":
            echo "$FRA[$texto]";
            break;
    } //fin switch
} //fin funcion

?>
```

2.4 Formulario

2.4.1 Formulario.php

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Formulario</title>
<script language="JavaScript" type="text/JavaScript">
<!--
function MM_jumpMenu(targ,selObj,restore){ //v3.0
    eval(targ+".location='"+selObj.options[selObj.selectedIndex].value+"'");
    if (restore) selObj.selectedIndex=0;
}
//-->
</script>
</head>

<body>

    <?php

        //me voy a conectar a la base de datos para recoger el valor m y obtener
        tambien la lista de fotos
        include("conexion.php");
        $consulta= " SELECT `m`
        FROM `lista`
        LIMIT 1 " ;

        $resultado= mysql_query($consulta , $conexion) or die(mysql_error());

        $sql_datos=mysql_fetch_array($resultado) or die("problema con lo del
        array");
        (integer)$m= $sql_datos["m"];
        mysql_close($conexion);
        include("cogelista.php");

        // $media sera el array con las fotos ordenadas segun sus medias
        include("media.php");

        $idioma=$_GET['idioma'];
        include("idiomas.php");
        ?>

<div align="center">
    <h2 align="center"><?php traduce("formulario",$idioma); ?>
    </h2>

    <br>
    <form method="post" action="iterativo.php">
        <table width="800" border="0" cellpadding="0" cellspacing="0">
            <tr>
                <td colspan="2"><p>
                    <?php traduce("form",$idioma); ?>
                </p>
            </td>
        </tr>
    </table>
    </form>

```

```

        <p>&nbsp;</p></td>
    </tr>
    <tr>
        <td><?php traduce("edad",$idioma); ?></td>
        <td><p>
            <select name="Edad" onChange="MM_jumpMenu(this,0)">
                <option value=0>&nbsp;</option>
                <option value="10">&lt;15</option>
                <option value=15>15-20</option>
                <option value=20>20-25</option>
                <option value=25>25-30</option>
                <option value=30>30-35</option>
                <option value=35>35-40</option>
                <option value=40>40-45</option>
                <option value=45>45-50</option>
                <option value=50>50-55</option>
                <option value=55>55-60</option>
                <option value=60>60-65</option>
                <option value="65">&gt;65</option>
            </select>
        </p></td>
    </tr>
    <tr>
        <td><?php traduce("sexo",$idioma); ?></td>
        <td><table width="200">
            <td><label>
                <input type="radio" name="Sexo" value="m">
                <?php traduce("masc",$idioma); ?>
            </label></td>
            <td><label>
                <input type="radio" name="Sexo" value="f">
                <?php traduce("fem",$idioma); ?>
            </label></td>
        </table></td>
    </tr>
    <tr>
        <td><?php traduce("nia",$idioma); ?></td>
        <td><p>
            <input name="NIA" type="text" maxlength="12">
        </p></td>
    </tr>
    <tr>
        <td><?php traduce("trast",$idioma); ?>
        <p>&nbsp;</p></td>
        <td><table width="200">
            <td><label>
                <input type="radio" name="Trast" value="si">
                <?php traduce("si",$idioma); ?>
            </label></td>
            <td><label>
                <input type="radio" name="Trast" value="no" checked >
                <?php traduce("no",$idioma); ?>
            </label></td>
        </table>
    </td>
    </tr>

```

```

        <p>&nbsp;</p></td>
    </tr>
</table>
<p>

</p>

<table width="800" border="0" cellspacing="2" cellpadding="0">
    <tr>
        <td></td>
        <td><?php
$ipOrigen = $_SERVER['REMOTE_ADDR'];
?>

        <input name="ip" type="hidden" value="<?php echo $ipOrigen; ?>" />
        <input name="idioma" type="hidden" value="<?php echo $idioma; ?>" />
        <!-- paso valores auxiliares para el funcionamiento del algoritmo
merge-sort -->
        <input name="a" type="hidden" value="0" />
        <input name="pulsado" type="hidden" value="0" />
        <input name="indice1" type="hidden" value="0" />
        <input name="indice2" type="hidden" value="1" />
        <input name="contador1" type="hidden" value="0" />
        <input name="contador2" type="hidden" value="0" />
        <input name="level" type="hidden" value="0" />
        <!-- paso uno a uno los valores del array-->
        <!-- paso los valores del array desordenado, que serán las fotos en un
future -->
        <?php $j=0 ?>
        <input name="primer" type="hidden" value="<?php echo $desorden[$j++];
?>" />
        <input name="segun" type="hidden" value="<?php echo $desorden[$j++];
?>" />
        <input name="tercer" type="hidden" value="<?php echo $desorden[$j++];
?>" />
        <input name="cuarto" type="hidden" value="<?php echo $desorden[$j++];
?>" />
        <input name="quinto" type="hidden" value="<?php echo $desorden[$j++];
?>" />
        <input name="sexto" type="hidden" value="<?php echo $desorden[$j++];
?>" />
        <input name="septimo" type="hidden" value="<?php echo
$desorden[$j++]; ?>" />
        <input name="octavo" type="hidden" value="<?php echo $desorden[$j];
?>" />
        <!-- paso los valores del array ordenado, incicalmente todo a cero -->
        <input name="primer1" type="hidden" value="0" />
        <input name="segun1" type="hidden" value="0" />
        <input name="tercer1" type="hidden" value="0" />
        <input name="cuartol" type="hidden" value="0" />
        <input name="quintol" type="hidden" value="0" />
        <input name="sextol" type="hidden" value="0" />
        <input name="septimol" type="hidden" value="0" />
        <input name="octavol" type="hidden" value="0" />

```

```

        <!-- Paso el valor "m". Indicativo de la lista de fotos que toca
        ahora, al final tendre que incrementar este valor
        y volver a mandarlo -->
        <input name="lista" type="hidden" value="<?php echo $m; ?>" />
        <!-- Paso las fotos ordenadas segun su puntuacion media -->
        <?php $j=0 ?>
        <input name="media1" type="hidden" value="<?php echo $media[$j++];
?>" />
        <input name="media2" type="hidden" value="<?php echo $media[$j++];
?>" />
        <input name="media3" type="hidden" value="<?php echo $media[$j++];
?>" />
        <input name="media4" type="hidden" value="<?php echo $media[$j++];
?>" />
        <input name="media5" type="hidden" value="<?php echo $media[$j++];
?>" />
        <input name="media6" type="hidden" value="<?php echo $media[$j++];
?>" />
        <input name="media7" type="hidden" value="<?php echo $media[$j++];
?>" />
        <input name="media8" type="hidden" value="<?php echo $media[$j]; ?>"
/>
        <div align="center"> <input type="submit" name="Submit" value="<?php
traduce("empezar",$idioma); ?>"></form>
</div>
        <p>&nbsp;</p>

<p>&nbsp;</p>
</div>
</body>
</html>

```

2.4.2 Conexion.php

```

<?php

$dbhost="localhost"; // host del MySQL (generalmente localhost)
$dbusuario=''; // aqui debes ingresar el nombre de usuario
                // para acceder a la base
$dbpassword=''; // password de acceso para el usuario de la
                // linea anterior
$db="DBGonzalo"; // Seleccionamos la base con la cual trabajar
$conexion = mysql_connect($dbhost, $dbusuario, $dbpassword) ;
if (! $conexion){

                                echo "Hubo un error al conectar
al servidor";

                                exit;

}

```

```
mysql_select_db($db, $conexion) or die("error al seleccionar la base de
datos");
```

```
?>
```

2.4.3 Cogelista.php

```
<?php
//coger la lista de un archivo de texto
//las listas estan en listas.txt
echo "<br>";
echo "<br>";

$vlneas = file("listas.txt");
$lista=$vlneas[$m];

$lista= explode("-", $lista);

for($i=0;$i<8;$i++){
$desorden[$i]=intval($lista[$i]);

}
?>
```

2.4.4 Media.php

```
<?php
//todo esto me va a sacar el array media de las fotos seleccionadasfoto

include("conexion.php");

$i=0;
$consulta= "SELECT `foto`,`media`
FROM `Fotos`
WHERE `foto` = ".$desorden[$i++]. "
OR `foto` = ".$desorden[$i++]. "
OR `foto` = ".$desorden[$i++]. "
OR `foto` = ".$desorden[$i++]. "
OR `foto` = ".$desorden[$i++]. "
OR `foto` = ".$desorden[$i++]. "
OR `foto` = ".$desorden[$i++]. "
LIMIT 0 , 30";

$resultado= mysql_query($consulta , $conexion) or die(mysql_error());

while ($sql_datos=mysql_fetch_array($resultado)){
```

```
$med[$sql_datos['foto']]=$sql_datos['media'];

}

mysql_close($conexion);
$i=0;
$j=0;
//el array media (de 0 a 7), contendra las fotos ordenadas de mayor a menor
segun su media
arsort($med);
$i=0;
foreach ($med as $key => $val) {

    $media[$i++]=$key;
}

?>
```

2.5 Iterativo

2.5.1 Iterativo.php

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>

<title>Estudio de la percepción de los colores en páginas Web</title>

<style type="text/css">
<!--
body {
    background-color: #888;
}
-->
</style></head>
<body>

<?php
//incluyo los idiomas
include("idiomas.php");
//recogo los datos
include("recogedatos.php");
?>
<?php
include("biblioteca.php");
```

```
ordena_calcula($ordenado,$desorden,$i,$j,$valor,$fase,$cont,$cont2,$level);
include("tiempo.php");
if ($fase==4){

include ("feedback.php");

exit;
}
```

2.5.2 Recogedatos.php

```
<?php
//tomo valores auxiliares a la funcion ordena_calcula
$fase= $_POST['a'];
$valor= $_POST['pulsado'];
$i=$_POST['indice1'];
$j=$_POST['indice2'];

(integer)$cont=$_POST['contador1'];
(integer)$cont2=$_POST['contador2'];
(integer)$level=$_POST['level'];

// tomo los valores del primer array, el desordenado, el que contiene las
fotos a comparar
(integer)$primero=$_POST['primer'];
(integer)$segun=$_POST['segun'];
(integer)$tercer=$_POST['tercer'];
(integer)$cuarto=$_POST['cuarto'];
(integer)$quinto=$_POST['quinto'];
(integer)$sexto=$_POST['sexto'];
(integer)$septimo=$_POST['septimo'];
(integer)$octavo=$_POST['octavo'];
//tomo los valores del segundo array, al principio estara a 0, lo iré
llenando poco a poco

(integer)$primero1=$_POST['primer1'];
(integer)$segun1=$_POST['segun1'];
(integer)$tercer1=$_POST['tercer1'];
(integer)$cuarto1=$_POST['cuarto1'];
(integer)$quinto1=$_POST['quinto1'];
(integer)$sexto1=$_POST['sexto1'];
(integer)$septimo1=$_POST['septimo1'];
(integer)$octavo1=$_POST['octavo1'];

//Tomo el valor m, que es la lista de imagenes que en ese momento toca

(integer)$lista=$_POST['lista'];

//tomo las fotos ordenadas segun sus medias

(integer)$media1=$_POST['media1'];
(integer)$media2=$_POST['media2'];
(integer)$media3=$_POST['media3'];
```



```
(integer)$media4=$_POST['media4'];
(integer)$media5=$_POST['media5'];
(integer)$media6=$_POST['media6'];
(integer)$media7=$_POST['media7'];
(integer)$media8=$_POST['media8'];

//Tomo los datos del usuario

$edad= $_POST['Edad'];
$sexo= $_POST['Sexo'];
$trast=$_POST['Trast'];
$ip=$_POST['ip'];

//tomo el idioma elegido
$id idioma=$_POST['idioma'];

//Tomo el Nia si lo hubiese
if (isset($_POST['NIA'])) {
$nia= $_POST['NIA'];
}

//tomo el valor del contador, cuenta las veces que muestra un par de
imagenes
if
(isset($_POST['contadorefotos'], $_POST['tiempomax'], $_POST['tiempomin'], $_POS
T['tiempotot'])) {

$contfoto=$_POST['contadorefotos'];
$tiempomax=$_POST['tiempomax'];
$tiempomin=$_POST['tiempomin'];
$tiempotot=$_POST['tiempotot'];
}
else {
    $contfoto=0;
    $tiempomax=0;
    $tiempomin=999999;
    $tiempotot=0;
}

// creo los arrays con los valores que me pasa la propia pagina o la
inicial:
$desorden=array($primero,$segun,$tercer,$cuarto,$quinto,$sexto,$septimo,$oct
avo);

$ordenado=array($primero1,$segun1,$tercer1,$cuarto1,$quinto1,$sexto1,$septim
o1,$octavo1);

$media=array($media1,$media2,$media3,$media4,$media5,$media6,$media7,$media8
);
?>
```

2.5.3 Mandatos.php

```

<input type="hidden" name="a" value="<?php echo $fase; ?> " />
<input type="hidden" name="indice1" value="<?php echo $i; ?>" />
<input type="hidden" name="indice2" value=" <?php echo $j; ?>" />
<?php $p=0 ?>
<input name="primer" type="hidden" value="<?php echo $desorden[$p++]; ?>" />
<input name="segun" type="hidden" value="<?php echo $desorden[$p++]; ?>" />
<input name="tercer" type="hidden" value="<?php echo $desorden[$p++]; ?>" />
<input name="cuarto" type="hidden" value="<?php echo $desorden[$p++]; ?>" />
<input name="quinto" type="hidden" value="<?php echo $desorden[$p++]; ?>" />
<input name="sexto" type="hidden" value="<?php echo $desorden[$p++]; ?>" />
<input name="septimo" type="hidden" value="<?php echo $desorden[$p++]; ?>"
/>
<input name="octavo" type="hidden" value="<?php echo $desorden[$p]; ?>" />

<!-- paso los valores del array ordenado -->

<?php $p=0 ?>
<input name="primer1" type="hidden" value="<?php echo $ordenado[$p++]; ?>"
/>
<input name="segun1" type="hidden" value="<?php echo $ordenado[$p++]; ?>" />
<input name="tercer1" type="hidden" value="<?php echo $ordenado[$p++]; ?>"
/>
<input name="cuarto1" type="hidden" value="<?php echo $ordenado[$p++]; ?>"
/>
<input name="quinto1" type="hidden" value="<?php echo $ordenado[$p++]; ?>"
/>
<input name="sexto1" type="hidden" value="<?php echo $ordenado[$p++]; ?>"
/>
<input name="septimo1" type="hidden" value="<?php echo $ordenado[$p++]; ?>"
/>
<input name="octavo1" type="hidden" value="<?php echo $ordenado[$p]; ?>" />

<!-- paso variables auxiliares -->

<input name="contador1" type="hidden" value="<?php echo $cont; ?>" />
<input name="contador2" type="hidden" value="<?php echo $cont2; ?>" />
<input name="level" type="hidden" value="<?php echo $level; ?>" />

<!-- Paso la variable m -->
<input name="lista" type="hidden" value="<?php echo $lista; ?>" />

<!-- Paso las fotos ordenadas segun su puntuacion media -->
<?php $p=0 ?>
<input name="media1" type="hidden" value="<?php echo $media[$p++]; ?>" />
<input name="media2" type="hidden" value="<?php echo $media[$p++]; ?>" />
<input name="media3" type="hidden" value="<?php echo $media[$p++]; ?>" />
<input name="media4" type="hidden" value="<?php echo $media[$p++]; ?>" />
<input name="media5" type="hidden" value="<?php echo $media[$p++]; ?>" />
<input name="media6" type="hidden" value="<?php echo $media[$p++]; ?>" />
<input name="media7" type="hidden" value="<?php echo $media[$p++]; ?>" />

```

```

<input name="media8" type="hidden" value="<?php echo $media[$p]; ?>" />

<!-- Paso los datos del usuario -->
<input name="Edad" type="hidden" value="<?php echo $edad; ?>" />
<input name="Sexo" type="hidden" value="<?php echo $sexo; ?>" />
<input name="Trast" type="hidden" value="<?php echo $trast; ?>" />
<input name="ip" type="hidden" value="<?php echo $ip; ?>" />

<!-- Paso el idioma elegido -->
<input name="idioma" type="hidden" value="<?php echo $idioma; ?>" />

<!-- Mando el contador -->
<input name="contadorfotos" type="hidden" value="<?php echo $contfoto; ?>" />

<!-- Paso medidas para el tiempo -->
<input name="tiempo" type="hidden" value="<?php echo $tiempo; ?>" />
<input name="tiempomin" type="hidden" value="<?php echo $tiempomin; ?>" />
<input name="tiempomax" type="hidden" value="<?php echo $tiempomax; ?>" />
<input name="tiempotot" type="hidden" value="<?php echo $tiempotot; ?>" />

<!-- mando el NIA en el caso que lo hubiese -->
<?php if (isset($_POST['NIA'])) {
    ?>
<input name="NIA" type="hidden" value="<?php echo $nia; ?>" />
<?php
}
?>
</form>

```

2.5.4 Biblioteca.php

```

<?php

Function
fase2(&$ordenado,&$desorden,&$i,&$j,&$valor,&$fase,&$cont,&$cont2,&$level){

// primero comparamos y ordenamos

    if ($valor==$desorden[(integer)$i]){
        if ($level==0){
            $a=($i+$j-2);
        }
        if ($level==1){
            $a=($i+$j-6);
        }

        $ordenado[(integer)$a]=$desorden[(integer)$i];
        $i=$i+1;
        $cont= $cont+1;
    }

    if ($valor==$desorden[(integer)$j]){
        if ($level==0){
            $a=($i+$j-2);

```

```

    }
    if ($level==1){
    $a=($i+$j-6);
    }

    $ordenado[(integer)$a]=$desorden[(integer)$j];
    $j=$j+1;
    $cont2= $cont2+1;

    }

    ////Si ya no hay que comparar mas, ordenamos

    if($cont==2 or $cont2==2){

    if ($cont==2 and $cont2==1)
        {
            $ordenado[(integer)$j]=$desorden[(integer)$j];
        }

    if ($cont==2 and $cont2==0)
        {
            $ordenado[(integer)$j]=$desorden[(integer)$j];
            $a=$j+1;
            $ordenado[(integer)$a]=$desorden[(integer)$a];
        }

    if ($cont==1 and $cont2==2)
        {
            $a=$i+2;
            $ordenado[(integer)$a]=$desorden[(integer)$i];

        }

    if($cont==0 and $cont2==2)
        {
            $a = $i+2;
            $ordenado[(integer)$a]=$desorden[(integer)$i];
            $a=$i+3;
            $b=$i+1;
            $ordenado[(integer)$a]=$desorden[(integer)$b];

        }
    if ($level==1)
    {
    $cont=0;
    $cont2=0;
    $i=0;
    $j=4;
    $fase=$fase+1;
    $desorden=$ordenado;

    }

```

```
if ($level==0)
{
$cont=0;
$cont2=0;
$i=4;
$j=6;
$level=$level+1;

}

////se prepara para recorrerlo una vez mas
}
}

function
fase3(&$ordenado,&$desorden,&$i,&$j,&$valor,&$fase,&$cont1,&$cont2,&$level){

//Si elige el valor del indice i
if ($valor==$desorden[(integer)$i])
{
$a=$i+$j-4;
$ordenado[(integer)$a]=$desorden[(integer)$i];
$i=$i+1;
if($i==4)
{

while ($j<8)
{
$ordenado[(integer)$j]=$desorden[(integer)$j];
$j++;
}

$fase=$fase+1;
$i=0;
$j=0;

}

}

// Si elige el valor del indice j
if ($valor==$desorden[(integer)$j])
{
$a=$i+$j-4;
$ordenado[(integer)$a]=$desorden[(integer)$j];
$a=$a+1;
$j=$j+1;
if($j==8)
{

while ($a<8)
{
$ordenado[(integer)$a]=$desorden[(integer)$i++];
$a++;
}
$fase=$fase+1;


```

```

    }

}

function
ordena_calcula (&$ordenado,&$desorden,&$i,&$j,&$valor,&$fase,&$cont1,&$cont2,
&$level){

    switch ($fase){
        case 0:
            $fase++;
            break;

        case 1:

if ($valor==$desorden[(integer)$i]){

    $ordenado[(integer)$i]=$desorden[(integer)$i];

    $ordenado[(integer)$j]=$desorden[(integer)$j];
    }
if ($valor==$desorden[(integer)$j]){

    $ordenado[(integer)$i]=$desorden[(integer)$j];

    $ordenado[(integer)$j]=$desorden[(integer)$i];
    }
    $i = $i+2;
    $j = $j+2;

        if ($j==9){

            $i=0;
            $j=2;

            $desorden=$ordenado;
            $fase=$fase+1;
        }

        break;

        //Fase 2, comparacion arrays 2 a 2

        case 2:

fase2($ordenado,$desorden,$i,$j,$valor,$fase,$cont1,$cont2,$level);
break;

        case 3:

fase3($ordenado,$desorden,$i,$j,$valor,$fase,$cont1,$cont2,$level);

```

2.5.5 Tiempo.php

2.5.6 Imagenes.php

up^{na}116
Universidad
Pública de Navarra
Nafarroako
Unibertsitate Publikoa
Todos los derechos reservados
Eskubide guztiak erresalbatu dira

```

<input type="hidden" name="pulsado" value="<?php echo
$desorden[(integer)$i]; ?>"><input type="image" src="imagenes/<?php echo
$imagen1; ?>" border='0' alt="foto 1"/><?php
include("mandadatos.php");?></form></td>
    <td width="300"><form action="iterativo.php" method="post"> <input
type="hidden" name="pulsado" value="<?php echo $desorden[(integer)$j];
?>"><input type="image" src="imagenes/<?php echo $imagen2; ?>" border='0'
alt="foto 2"/></td>
    <td width="100">&nbsp;</td>
</tr>
</table>
<p align="center"><font size=4><?php traduce("imagen1",$idioma); ?></p>
<p align="center"><?php traduce("imagen2",$idioma); ?></font></p>

```

2.6 Feedback

2.6.1 Feedback.php

```

<?php

$tiempomedio=round($tiempotot/$contfoto,2);

$nuevo=$lista+1;
if ($nuevo==4000){
    $nuevo=1;
}
$foto1=$ordenado[0];
$foto2=$ordenado[1];
$foto3=$ordenado[2];
$foto4=$ordenado[3];
$foto5=$ordenado[4];
$foto6=$ordenado[5];
$foto7=$ordenado[6];
$foto8=$ordenado[7];

include("conexion.php");

//cogo las vistas que lleva cada foto en el array vistas, y los puntos que
lleva cada foto en el array puntos
include("vistasypuntos.php");
include("meter.php");

//Actualizo en la base de datos el valor m, que indica la lista siguiente
que mostrara al proximo usuario
$consulta= "UPDATE `DBGonzalo`.`lista` SET `m` = '$nuevo' WHERE `lista`.`m`
= $lista LIMIT 1";

$resultado= mysql_query($consulta , $conexion) or die(mysql_error());

//Meto los datos del Usuario
include("infousuario.php");
//$id es la variable id del infousuarios
//Meto la seleccion del usuario

```



```
include("seleccionusuario.php");
mysql_close($conexion);
//aqui viene donde se muestran las imagenes elegidas por el usuario y segun
su media
?>
<!-- El array ordenado segun su media es:<br /> -->
<?php

include("nuevofeed.php");

?>
```

2.6.2 Vistasypuntos.php

```
<?php

$consulta="SELECT `foto`,`vistas`,`puntos` FROM `Fotos` WHERE
`foto`=$ordenado[0] OR `foto`=$ordenado[1] OR `foto`=$ordenado[2] OR
`foto`=$ordenado[3] OR `foto`=$ordenado[4] OR `foto`=$ordenado[5] OR
`foto`=$ordenado[6] OR `foto`=$ordenado[7]";

$resultado= mysql_query ($consulta, $conexion) or die(mysql_error());
$i=0;
while ($row = mysql_fetch_array ($resultado))
{

    $vistas[$row['foto']]=$row['vistas'];
    $puntos[$row['foto']]=$row['puntos'];
    $i++;

}

?>
```

2.6.3 Meter.php

```
<?php

//aqui calculo los nuevos valores para las vistas y los puntos, y la nueva
media
for($i=0;$i<8;$i++){
    $vistas[$ordenado[$i]]=$vistas[$ordenado[$i]]+1;
}
for($i=0;$i<8;$i++){
    $puntos[$ordenado[$i]]=$puntos[$ordenado[$i]]+(8-$i);
}

for($i=0;$i<8;$i++){
    $nuevamedia[$ordenado[$i]]=round($puntos[$ordenado[$i]]/$vistas[$ordenado[$i]]
,2);
}
//aqui introduzco los valores a la base de datos
for($j=0;$j<8;$j++){
```

```

$i=$ordenado[$j];
$consulta = "UPDATE `DBGonzalo`.`Fotos` SET `vistas` = $vistas[$i] ,
`puntos` = $puntos[$i], `media` = $nuevamedia[$i] WHERE `Fotos`.`foto` = $i
";

$resultado= mysql_query ($consulta, $conexion) or die(mysql_error());
}
?>

```

2.6.4 Infousuario.php

```

<?php

$fecha=date('d-m-Y');
$consulta2= "INSERT INTO `DBGonzalo`.`Infousuarios` (
`id` ,
`m` ,
`Edad` ,
`Sexo` ,
`ip` ,
`NIA` ,
`trast` ,
`fecha` ,
`idioma`
)
VALUES (
NULL , '$nuevo', '$edad', '$sexo', '$ip', '$nia' , '$trast',
'$fecha', '$idioma'
)";
$resultado= mysql_query($consulta2 , $conexion) or die(mysql_error());

$consulta3= "SELECT *
FROM `Infousuarios`
ORDER BY `id` DESC
LIMIT 1
";
$resultado= mysql_query($consulta3 , $conexion) or die(mysql_error());
$sql_datos=mysql_fetch_array($resultado) or die("problema con lo del
array");
(integer)$id=$sql_datos["id"];

?>

```

2.6.5 Seleccionsusuario.php

```

<?php
$consulta4="INSERT INTO `DBGonzalo`.`SeleccionUsuario` (
`id` ,
`foto1` ,
`valor1` ,
`foto2` ,
`valor2` ,
`foto3` ,
`valor3` ,

```

```

`foto4` ,
`valor4` ,
`foto5` ,
`valor5` ,
`foto6` ,
`valor6` ,
`foto7` ,
`valor7` ,
`foto8` ,
`valor8` ,
`tiempotot` ,
`tiempomin` ,
`tiempomed` ,
`tiempomax`
)
VALUES (
'id', '$foto1', '8', '$foto2', '7', '$foto3', '6', '$foto4', '5',
'$foto5', '4', '$foto6', '3', '$foto7', '2', '$foto8', '1', '$tiempotot' ,
'$tiempomin' , '$tiempomedio' , '$tiempomax'
)";

$resultado= mysql_query($consulta4 , $conexion) or die(mysql_error());
?>

```

2.6.6 Nuevofeed.php

```

<?php
$vlneas = file("fotos.txt");

for($i=0;$i<8;$i++){
    $foto[$i]=$vlneas[(integer)$ordenado[$i]];
}
for($i=0;$i<8;$i++){
    $fotoM[$i]=$vlneas[(integer)$media[$i]];
}

$i=0;
$j=0;
?>

<p align="center">&nbsp;</p>
<p align="center">&nbsp;</p>
<p align="center"><font size=5>
    <?php traduce("agradecer",$idioma); ?>
</font></p>
<p>&nbsp;</p>
<p>&nbsp;</p>
<table width="604" border="0">
    <tr>
        <td width="108">&nbsp;</td>
        <td width="585"><font size=4><?php traduce("feed1",$idioma);
?></font></td>

    </tr>

```

```

</table>
<p></p>
<table width="885" height="145" border="0" cellpadding="0" cellspacing="20">
  <tr>
    <td width="100"> </td>
    <td width="103"><input name="pulsado" type="image" src="imagenes/<?php
echo $foto[$i++]; ?>" width="103" height="77" /></td>
    <td width="103"><input name="pulsado" type="image" src="imagenes/<?php
echo $foto[$i++]; ?>" width="103" height="77" /></td>
    <td width="103"><input name="pulsado" type="image" src="imagenes/<?php
echo $foto[$i++]; ?>" width="103" height="77" /></td>
    <td width="103"><input name="pulsado" type="image" src="imagenes/<?php
echo $foto[$i++]; ?>" width="103" height="77" /></td>
    <td width="103"><input name="pulsado" type="image" src="imagenes/<?php
echo $foto[$i++]; ?>" width="103" height="77" /></td>
    <td width="103"><input name="pulsado" type="image" src="imagenes/<?php
echo $foto[$i++]; ?>" width="103" height="77" /></td>
    <td width="103"><input name="pulsado" type="image" src="imagenes/<?php
echo $foto[$i++]; ?>" width="103" height="77" /></td>
  </tr>
</table>

<p>&nbsp;</p>
<table width="604" border="0">
  <tr>
    <td width="107">&nbsp;</td>
    <td width="581"><font size=4><?php traduce("feed2",$idioma);
?></font></td>

  </tr>
</table>
<p></p>
<table width="885" height="145" border="0" cellpadding="0" cellspacing="20">
  <tr>
    <td width="100"> </td>
    <td width="103"><input name="pulsado" type="image" src="imagenes/<?php
echo $fotoM[$j++]; ?>" width="103" height="77" /></td>
    <td width="103"><input name="pulsado" type="image" src="imagenes/<?php
echo $fotoM[$j++]; ?>" width="103" height="77" /></td>
    <td width="103"><input name="pulsado" type="image" src="imagenes/<?php
echo $fotoM[$j++]; ?>" width="103" height="77" /></td>
    <td width="103"><input name="pulsado" type="image" src="imagenes/<?php
echo $fotoM[$j++]; ?>" width="103" height="77" /></td>
    <td width="103"><input name="pulsado" type="image" src="imagenes/<?php
echo $fotoM[$j++]; ?>" width="103" height="77" /></td>
    <td width="103"><input name="pulsado" type="image" src="imagenes/<?php
echo $fotoM[$j++]; ?>" width="103" height="77" /></td>
    <td width="103"><input name="pulsado" type="image" src="imagenes/<?php
echo $fotoM[$j++]; ?>" width="103" height="77" /></td>
  </tr>

```

```

</table>
<p>

</p>
<p>&nbsp;</p>
<p>&nbsp;</p>
<?php
include("repetir.php");
?>

```

2.6.6.1 Repetir.php

```

<?php

// Botones para repetir el test o volver a la página del departamento
// Para repetir el test mando los mismos datos de usuario y los datos
iniciales
// necesarios para iterativo.php

//me voy a conectar a la base de datos para recoger el valor m y obtener
tambien la lista de fotos
include("conexion.php");
$consulta= " SELECT `m`
FROM `lista`
LIMIT 1 " ;

$resultado= mysql_query($consulta , $conexion) or die(mysql_error());

$sql_datos=mysql_fetch_array($resultado) or die("problema con lo del
array");
(integer)$m= $sql_datos["m"];
mysql_close($conexion);
include("cogelista.php");
// $media sera el array con las fotos ordenadas segun sus medias
include("media.php");
?>
<div align="center"><em><strong><?php traduce("rep1",$idioma);
?></strong></em></div>
<table width=100% border="0" cellspacing="2" cellpadding="0">
<tr>
<td width=20%>&nbsp;</td>
<td width=62%>
<form method="GET" action=""formulario.php">

<input name="idioma" type="hidden" value=""<?php echo
$idioma; ?>" />
<em> <strong><?php traduce("rep2",$idioma); ?></strong></em></p>
<p>
<input type="submit" name="Submit" value=""<?php traduce("rep3",$idioma);
?>">

</p>
</form>
<p>
<form action="../../index.php">

```

```

<p><em><strong><?php traduce("rep4",$idioma); ?></strong></em></p>
<p>
<input type="submit" name="Submit" value="<?php traduce("rep5",$idioma);
?>">

</p>
</form></td>
<td width=18%>&nbsp;</td>
</tr>
</table>

```

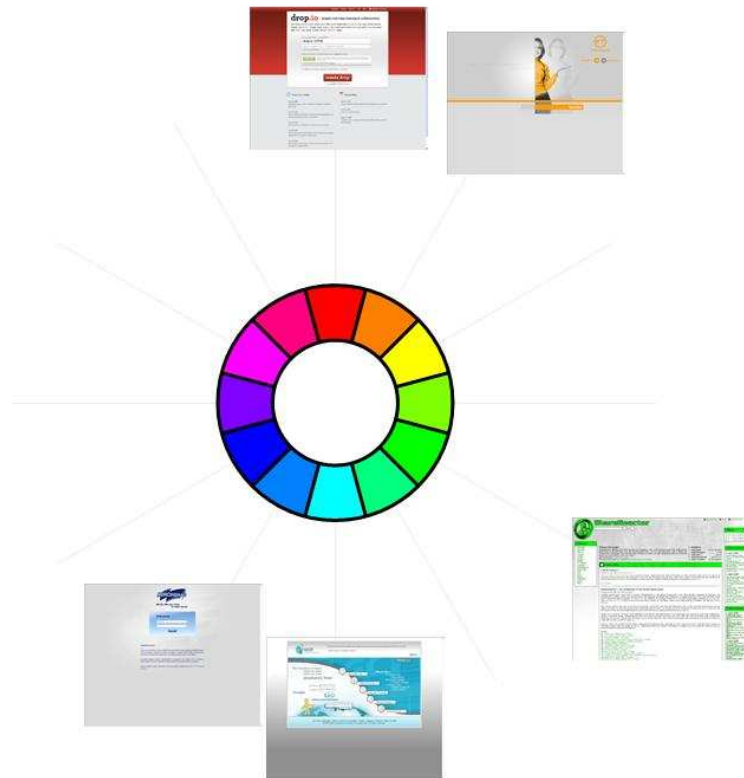
3 Otros criterios de selección de Imágenes

A la hora de seleccionar las imágenes, se emplearon otra serie de criterios, con el fin de abarcar el mayor rango de combinaciones cromáticas. Estos diferentes criterios podrán ser usados en el futuro para diferentes estudios estadísticos.

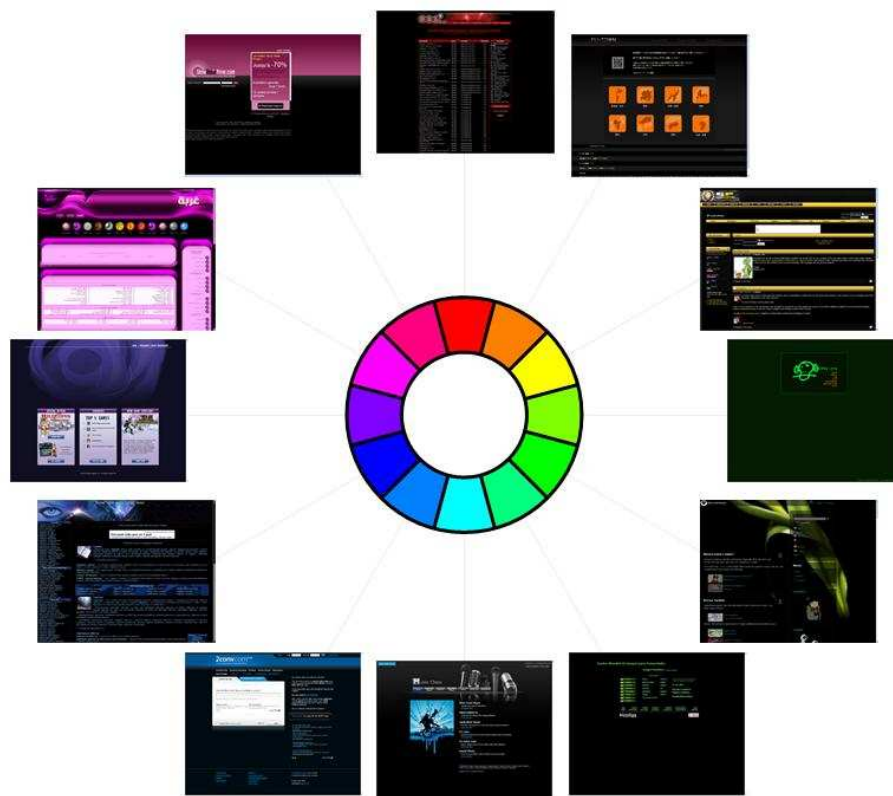
3.1 Monocromáticas



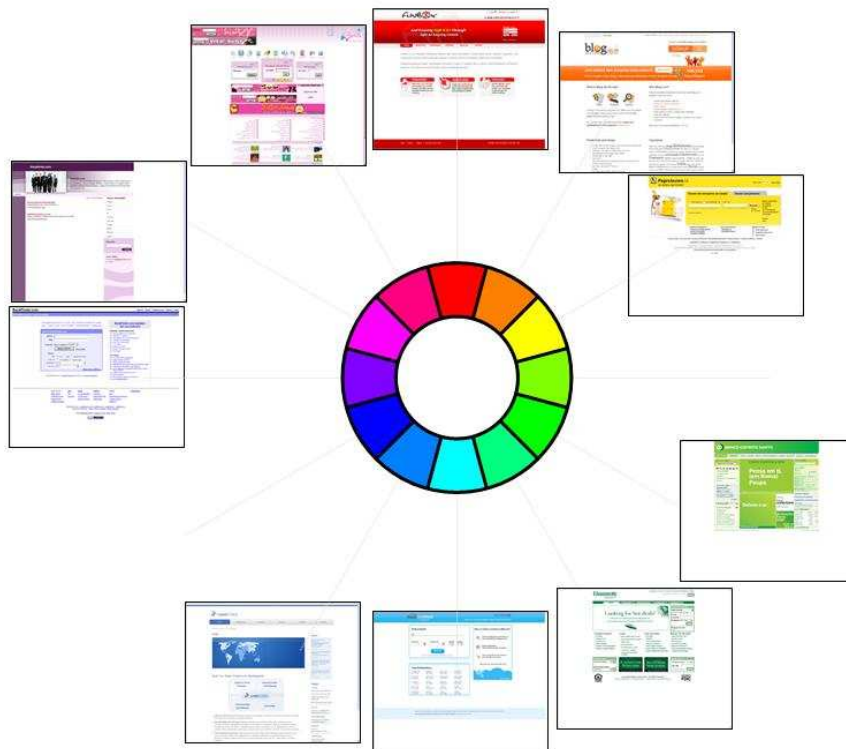
3.2 Grises con diferentes tonos



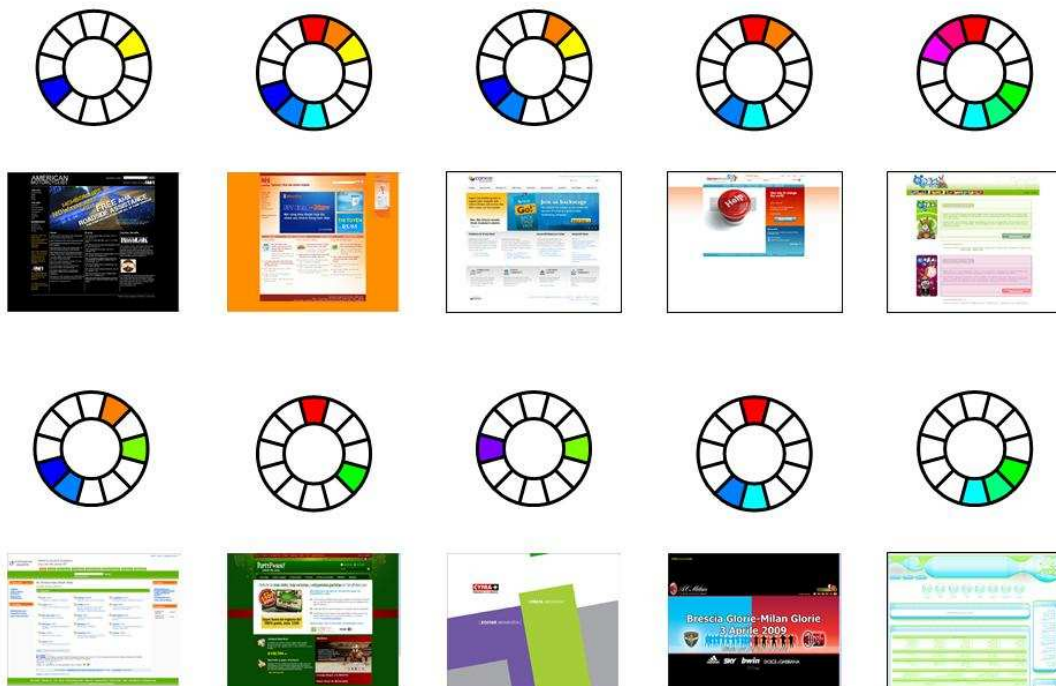
3.3 Negras y tonos



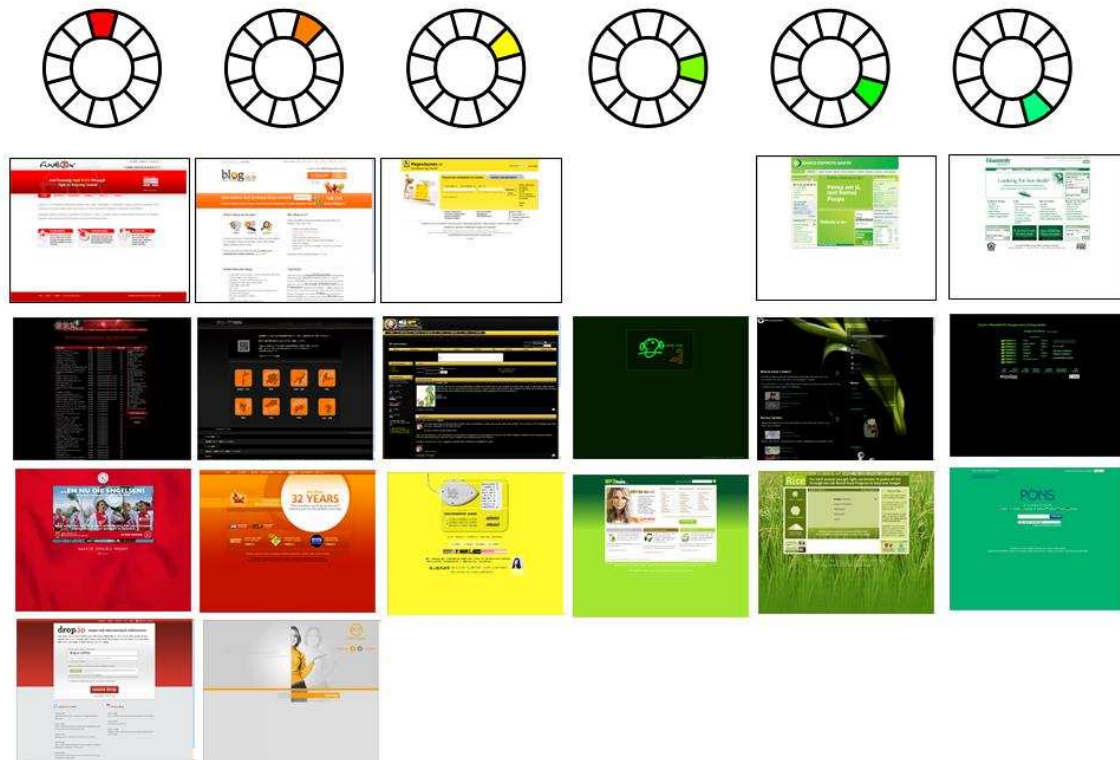
3.4 Blancas y tonos



3.5 Tonos opuestos



3.6 Tonos Cálidos



3.7 Tonos Fríos

